AD-A048 840 MCDONNELL DOUGLAS ASTRONAUTICS CO HUNTINGTON BEACH CALIF F/G 1/1
AERODYNAMIC COMPUTER CODE FOR COMPUTING PRESSURE LOADING ON COM--ETC(U)
JAN 78 K K WANG
UNCLASSIFIED MDC-G7215 NL

1 OF 3
ADA048840

12

MDC G7215 NW

6

AERODYNAMIC COMPUTER CODE FOR COMPUTING PRESSURE LOADING ON COMPLETE MISSILE FOR STRUCTURAL ANALYSIS

10

Prepared by Kenneth K. Wang

MCDONNELL DOUGLAS ASTRONAUTICS COMPANY
5301 BOLSA AVENUE
HUNTINGTON BEACH, CALIFORNIA 92647

14 MDC-G7215

11 JAN[redacted]978 12 248 p.

9 Final rept. 1 Jun 76-31 Dec 77,

15 FINAL REPORT
CONTRACT NUMBER N60921-76-C-0164
JUNE 1, 1976 TO DECEMBER 31, 1977

Approved for Public Release
Distribution Unlimited

Prepared for
NAVAL SURFACE WEAPONS CENTER
WHITE OAK, SILVER SPRING, MARYLAND 20910

This technical report has been reviewed and is approved.

P. C. Huang
Naval Surface Weapons Center

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER MDC G7215	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Aerodyanmic Computer Code for Computing Pressure Loading on Complete Missile for Structural Analysis		5. TYPE OF REPORT & PERIOD COVERED Final Report June 1, 1976 to December 31, 1977
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Kenneth K. Wang		8. CONTRACT OR GRANT NUMBER(s) N60921-76-C-0164 new
9. PERFORMING ORGANIZATION NAME AND ADDRESS McDonnell Douglas Astronautics Company 5301 Bolsa Avenue Huntington Beach, California 92647		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Surface Weapons Center White Oak Silver Spring, Maryland 20910		12. REPORT DATE January 1978
		13. NUMBER OF PAGES 212
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

For public release, distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Aerodynamics
Pressure distribution on missile or its components
Structural analysis

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Aerodynamic computer code for calculating the pressure distribution on missile or its components e.g., body and wings and to interpolate by surface fit at locations as specified for structural analysis using the NASTRAN computer code.

PREFACE

This document reports the work performed by McDonnell Douglas Astronautics Company for the Naval Surface Weapons Center, White Oak Laboratory, Silver Spring, Maryland for the development of an aerodynamic code for computing pressure loads on a missile and/or its components. The work was accomplished under contract number N60921-76-C-0164. The Naval Sea Systems Command sponsored the activity.

An MDAC aerodynamic code was extensively modified to accept missile geometry and finite element descriptions from the NSWC computer codes PING and BING to compute pressure loading on the missile and to interpolate by surface fit the pressure load at specified locations in a NASTRAN bulk data format.

The theoretical background of the method is described briefly in this report. Computer code organization and input parameters are described in detail. For the user's convenience, test cases and instructions are included.

CONTENTS

LIST OF SYMBOLS

k, K	Interpolation coefficients
u, v, w	Perturbation velocities
x, y, z	Coordinates
ξ, η	Integration variables
ϕ	Potential functions
L	Slope of panel sweep, $\tan \Lambda$
M_∞	Free-stream Mach number
N	Number of data points to be interpolated
P	Pressure load
r	Radial distance
δ	Deflection

SUMMARY

This report represents the work performed under contract number N60921-76-C-0164 for the Naval Surface Weapons Center, White Oak Laboratory, Silver Spring, Maryland under Naval Sea Systems Command sponsorship for the development of an aerodynamic code for computing distributed pressure loads on a missile or its components such as wings, body or tails.

Major modifications were made to the existing MDAC aerodynamic code to streamline the input format, restructure the code and reduce the computer core memory requirement. In addition, new subroutines were developed as necessary to provide versatility in the input specification, efficient computation and informative outputs. The input parameters have been reduced to simplify input preparation. Automated generation of wing camber, thickness and twist directly from the wing surface description has been incorporated. Allowances were made in specifying the finite element in a coordinate system other than that used for aerodynamic calculations. At the completion of aerodynamic analysis, the code generates the pressure distribution on the missile at the centroid of each finite element as specified. They are produced as punched card output in a NASTRAN bulk card format as desired.

Section 1
INTRODUCTION

The design and analysis of flight vehicles requires coordinated, joint efforts among various technical disciplines such as structures, aerodynamics, flight kinetics, and propulsion. For example, aerodynamic calculations provide the loading to be used in the structural analyses, and configuration changes caused by bending, erosion, damage, etc., necessitate renewed aerodynamic calculations. With the development of high-speed, large-capacity computers, the analysis performed for individual disciplines has been highly automated into computing codes capable of analyzing very complex configurations by appropriate modeling for structures, notably NASTRAN (Reference 1), and various finite-element, lifting-surface-theory codes for aerodynamics (References 2 and 3). Since these codes were developed independently of each other, and major physical modeling differences exist among disciplines, some interfacing tasks must be performed when the output of one code is supplied as input of the other. These tasks have been performed manually in the past and can become cumbersome if the analysis requires numerous iterations between the disciplines. It is therefore both desirable and necessary to develop an aerodynamics code to interface with NSWC finite-element generating code PING or BING (References 4 and 5) and NASTRAN structure codes as an initial step toward a more automated design procedure.

A variety of aerodynamics codes applicable to different speed regimes and vehicle configurations are available at MDAC. For present purposes it was decided to modify an existing MDAC finite-element code, which is based on Woodward's method of aerodynamic influence coefficients (Reference 3). As opposed to other available finite-element methods of aerodynamic prediction (Reference 2), Woodward's method has the unique capability of being applicable in both the subsonic and low-supersonic ($M_\infty < 3$) speed regimes. Although the method is theoretically deficient at transonic Mach numbers, predictions for wing-body combinations have been shown to be acceptable. A detailed description of the method will be found in Section 2.

At the conclusion of aerodynamic calculations, the present code generates the pressure loads on a missile (body, wings and nacelles) at the control point of surface panels. Interpolation and rearrangement of the pressure load to suit structural analysis will be required because, in general, placement of the panel is mainly governed by aerodynamic considerations. For this purpose, an efficient and versatile interpolation method is selected.

Many factors entered into the selection of the surface spline fit method for interpolating the pressure loads. In the present case, the method based on the differential equation relating deflection and load of thin plates (Reference 6) was considered to be the most suitable one. It requires less core storage and is extremely simple to code. Furthermore, the method accepts input data either in a regular form or randomly distributed.

The final output of the code is the pressure load at the centroid of each finite element. For structural analysis, these data are produced as punched cards in the NASTRAN bulk data format.

It may be mentioned that this described approach was employed in an initial effort in the development of a MDAC aero code for interface with the NSWC finite-element code PING for pressure loads on missile wings (contract number N60921-75-C-0069). The success achieved and the experience gained in the effort have been most helpful to the present work. In particular, improvements such as input simplification and coordinate system flexibility have been incorporated as a direct result of this earlier effort.

Section 2
AERODYNAMIC THEORY AND SURFACE SPLINE FIT METHOD

The MDAC aerodynamic code modified for computing pressure loads on a missile for structural analysis is based on the method of influence coefficients developed by Woodward (Reference 3). The code provides the pressure load at the control point of each panel as defined in the program. For structural analysis the pressure distribution needs to be interpolated at the centroid of each finite element. An efficient surface spline method is employed for this purpose.

In the following subsections, both the aerodynamic theory and the surface spline method are discussed.

2.1 AERODYNAMIC THEORY

For small perturbations, the governing equation of potential flow can be simplified greatly and solution of many problems becomes possible. The present code solves the linearized, three-dimensional potential equation

$$\left(1 - M_\infty^{\ 2}\right) \phi_{xx} + \phi_{yy} + \phi_{zz} = 0 \qquad (1)$$

derived from the exact equations of motion by neglecting all terms containing squares of higher powers of the perturbation velocities. Except for the case of transonic flow ($M_\infty \sim 1$), Equation 1 is valid for both subsonic ($M_\infty \lesssim 1$) or supersonic flows ($1 < M_\infty < 3$). For hypersonic flow ($M_\infty \gg 3$), the equation requires additional terms to be applicable. It should be noted here that the coordinate system used for aerodynamic analysis follows the accepted convention with the x-axis in the direction of flow, z-axis positive upwards and y-axis completing the right-hand system.

Considering Equation 1, a number of fundamental solutions exist that can be used to solve the problem of computing pressures, forces, and moments over the wing and body of a missile. Physically speaking, they are known as

source, doublet, and vortex singularities. In the present treatment, the effect of body volume, incidence and camber are represented by line sources and doublets distributed along the body axis; the effects of wing thickness are represented by planar source distributions. Wing camber, twist, and angle-of-attack effects are represented by vortex distributions. As Equation 1 is linear, the sum of these fundamental solutions is also a solution. It only remains necessary to determine their strengths to satisfy the boundary condition on the body and wing surface, i.e., flow must be tangent to the surface at selected points. The interference effect of wing on the body is accounted for by additional vortex distributions on the body panel surface.

The fundamental solutions of the linear equation are well known (e.g., Reference 3). For instance, in integral form, the potential for line source is

$$\phi_1 = -\int \frac{\xi d\xi}{\left[(x-\xi)^2 + \left(1-M_\infty^{\ 2}\right) r^2\right]^{1/2}} \tag{2}$$

The potential for line doublets is

$$\phi_2 = \int \frac{\sin\theta \ \xi \ (x-\xi) d\xi}{\left[(x-\xi)^2 + \left(1-M_\infty^{\ 2}\right) r^2\right]^{1/2} r} \tag{3}$$

The potential for a constant source distribution is

$$\phi_3 = -\frac{C_1}{\pi} \iint \frac{d\xi \ d\eta}{\left\{(x-\xi)^2 + \left(1-M_\infty^{\ 2}\right)\left[(y-\eta)^2 + z^2\right]\right\}^{1/2}} \tag{4}$$

The potential for a linearly varying source distribution is

$$\phi_4 = \frac{C_1}{\pi} \iint \frac{(L\eta-\xi) \ d\xi \ d\eta}{\left\{(x-\xi)^2 + \left(1-M_\infty^{\ 2}\right)\left[(y-\eta)^2 + z^2\right]\right\}^{1/2}} \tag{5}$$

For constant pressure vortex distributions, the potential is

$$\phi_5 = \frac{C_1}{\pi} \iint \frac{z(x-\xi)\, d\xi\, d\eta}{\left[(y-\eta)^2 + z^2\right] \left\{(x-\xi)^2 + \left(1-M_\infty^2\right)\left[(y-\eta)^2 + z^2\right]\right\}^{1/2}}$$

$$+ \frac{1-C_1}{\pi} \iint \frac{z\, d\xi\, d\eta}{(y-\eta)^2 + z^2} \qquad (6)$$

where

$C_1 = 1$ for supersonic flow, and $C_1 = 0.5$ for subsonic flow.

Integrating Equation 2 between limits $\xi = 0$ and $\xi = \ell$ as shown in Figure 1 yields the potential due to a finite line source of unit strength.

$$\phi_1 = \mathrm{Re}\left[d_1 - d_2\right] - x_1(F_1 - F_2) \qquad (7)$$

CR103

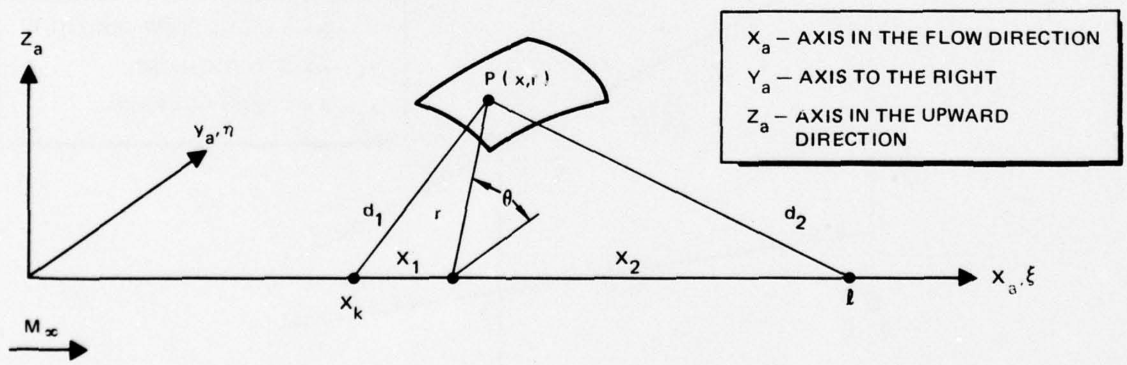


Figure 1. Line Source or Doublet Geometry

For line doublets of unit strength, the potential obtained from Equation 3 after integration is

$$\phi_2 = \frac{\sin\theta}{2r} \mathrm{Re}\left[x_1(d_1 - d_2) - \ell d_2 + r^2 \left(1-M_\infty^2\right) \log \frac{x_1 + d_1}{x_2 + d_2}\right] \qquad (8)$$

where

$$d_1 = \left[(x-x_k)^2 + \left(1-M_\infty^2\right) r^2\right]^{1/2}$$

$$d_2 = \left[(x-\ell)^2 + \left(1-M_\infty^2\right) r^2\right]^{1/2}$$

$$F_1 = \text{Re} \log (x-x_k + d_1)/ \left(\left|1-M_\infty^2\right|\right)^{1/2} r$$

$$F_2 = \text{Re} \log (x-\ell + d_2)/ \left(\left|1-M_\infty^2\right|\right)^{1/2} r$$

r = radial distance

θ = angle between r and x-y plane

For source distribution over a panel, the limits of integration for Equations 4, 5 and 6 are defined by the boundaries of the panel as shown in Figure 2.

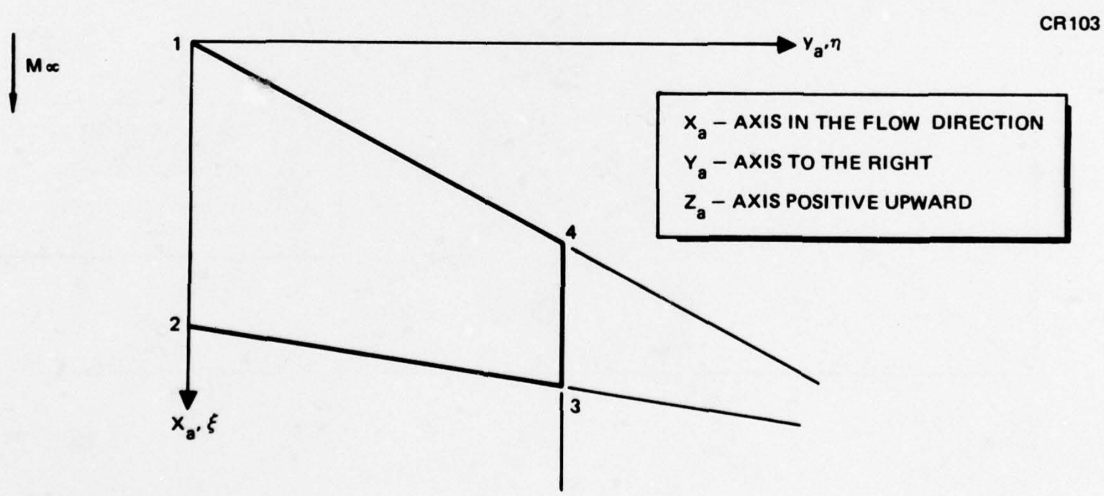


Figure 2. Source Panel Geometry

The above integrals can be considered as the sum of four integrals each representing a semi-infinite triangular region having origins at each of the corners. In other words, we have the potential for a particular panel as

$$\phi = \sum_{i=3}^{5} (\phi_{i1} - \phi_{i2} - \phi_{i3} + \phi_{i4}) \tag{9}$$

where ϕ_{ik} is the component potential of the semi-infinite triangular region with the k-th corner as its origin. The limit of integration for each region can easily be found (Reference 3) and the integrals subsequently evaluated. For a constant source, a linearly varying source and constant pressure vortex distributions, the component potentials are

$$\phi_{3k} = -\frac{C_1}{\pi}\left[yF_3 + (x-Ly)\,F_4 - zF_5\right] \tag{10}$$

$$\begin{aligned}\phi_{4k} = -\frac{C_1}{2\pi}\ \Big\{&\left[2xy + L\,(z^2 - y^2)\right]F_3 \\ &+ \left[(x-Ly)^2 - L^2 + 1-M_\infty{}^2\right]F_4 - 2z(x-Ly)\,F_5 - yd\Big\}\end{aligned} \tag{11}$$

$$\phi_{5k} = C_1\Delta u\ \left\{(x-Ly)(F_5 + F_6) + z\left[L^2 + 1-M_\infty{}^2\right]F_4 - L(F_3-F_7)\right\}/2\pi \tag{12}$$

where

$$F_3 = R_e\left\{\log(x + d)/\left[\left|1-M_\infty{}^2\right|r\right]^{1/2}\right\}$$

$$F_4 = R_e\left\{\log(x' + d')/\left[\left|1-M_\infty{}^2\right|r'\right]^{1/2} / \left(L^2 + 1-M_\infty{}^2\right)^{1/2}\right.$$

$$F_5 = R_e\left[\tan^{-1} zd/(Lr-xy)\right]$$

$$\begin{aligned}F_6 &= \tan^{-1}(y/z) \text{ for } M_\infty < 1 \\ &= 0 \text{ for } M_\infty \geq 1\end{aligned}$$

$$\begin{aligned}F_7 &= \log\,(Lr/r') \text{ for } M_\infty \leq 1 \\ &= 0 \text{ for } M_\infty > 1\end{aligned}$$

and

$$r = \left(y^2 + z^2\right)^{1/2}$$

$$d = \left[x^2 + \left(1 - M_\infty^{\,2}\right) r^2\right]^{1/2}$$

$$x' = Lx + \left(1 - M_\infty^{\,2}\right) y$$

$$y' = x - Ly$$

$$r' = \left[(x - Ly)^2 + \left(L^2 + 1 - M_\infty^{\,2}\right)^2\right]^{1/2}$$

$$d' = \left\{ \left[Lx + \left(1 - M_\infty^{\,2}\right) y\right]^2 + \left(1 - M_\infty^{\,2}\right)\left[(x - Ly)^2 + \left(L^2 + 1 - M_\infty^{\,2}\right) z^2\right] \right\}^{1/2}$$

Δu = axial velocity discontinuity = $- 1/2 \ \Delta Cp$

R_e = real part

The velocity components are obtained from the potentials by differentiation

$$u = \frac{\partial \phi}{\partial x}$$

$$v = \frac{\partial \phi}{\partial y}$$

$$w = \frac{\partial \phi}{\partial z}$$

Applying the tangency condition to the control point of each panel, a system of linear algebraic equations is formed with the strength of singularities of each panel constituting an unknown which is to be determined. The location of the control point (i. e., the point where the pressure is computed for each panel) has been determined empirically to be 95 percent of the local panel chord through the centroid. This selection has been correlated with extensive numerical computations of pressure distributions and lift curve slopes for a

variety of wing planforms at both subsonic and supersonic speeds. By relating the strength of singularities to the velocity components, the solution of this system of equations yields a coefficient matrix of influence coefficients which are to be used for computing the surface pressure, lift, drag, and pitching moments.

The pressure coefficients on the body resulting from line sources and doublets are

$$C_{PBBi} = -2\bar{U}_{Bi} + \beta^2\bar{U}_{Bi}^{\ 2} - \bar{V}_{Bi}^{\ 2} - \bar{W}_{Bi}^{\ 2} \tag{13}$$

where $\bar{U}_{Bi}$, $\bar{V}_{Bi}$ and $\bar{W}_{Bi}$ are velocity components on body panel i induced by the sources and doublets. The pressure coefficients on the body panel resulting from surface distributions of singularities on the body and wing are

$$C_{PBWi} = -2\hat{U}_{Bi} + \beta^2\hat{U}_{Bi}^{\ 2} - \hat{V}_{Bi}^{\ 2} - \hat{W}_{Bi}^{\ 2} \tag{14}$$

where $\hat{U}_{Bi}$, $\hat{V}_{Bi}$ and $\hat{W}_{Bi}$ are velocity components on body panel i induced by body and wing panels. The combined pressure coefficient C_{PBi} on the body panel i in the presence of wing is the sum of these two coefficients, C_{PBBi} and C_{PBWi}.

The pressure coefficients on the wing panels are

$$C_{PWi} = -2U_{Wi} + \beta^2U_{Wi}^{\ 2} - V_{Wi}^{\ 2} - W_{Wi}^{\ 2} \tag{15}$$

where

$U_{Wi} = U_{WBi} + U_{WWi}$, $V_{Wi} = V_{WBi} + V_{WWi}$ and

$W_{Wi} = W_{WBi} + W_{WWi}$ are the velocity components due to the presence of body and wing panels.

The lift, drag, and pitching moment coefficients are obtained by summing the contribution from all panels.

$$C_L = \frac{1}{A_T}\left\{\sum_i^{NC}\sum_j^{NB}\left[C_{PW}A\right]_{ij} + \sum_i^{NX}\sum_j^{NT}\left[C_{PB}A\right]_{ij}\right\} \quad (16)$$

$$C_D = \frac{1}{A_T}\left\{\sum_i^{NC}\sum_j^{NB}\left[C_{PW}\left(\frac{dz}{dx}-\alpha\right)A\right]_{ij}\right.$$

$$\left. + \sum_i^{NX}\sum_j^{NT}\left[C_{PB}\left(\frac{dz}{dx}-\alpha\right)A\right]_{ij}\right\} \quad (17)$$

$$C_M = \frac{1}{A_T\bar{c}}\sum_i^{NC}\sum_j^{NB}\left[C_{PW}A\left\{x_i + z_i\left(\frac{dz}{dx}-\alpha\right)\right\}\right]_{ij}$$

$$+ \frac{1}{A_T\bar{c}}\sum_i^{NX}\sum_j^{NT}\left[C_{PB}A\left\{x_i + z_i\left(\frac{dz}{dx}-\alpha\right)\right\}\right]_{ij} \quad (18)$$

where

C_{pB}, C_{pW}	=	Pressure coefficient for body or wing panel
A_T	=	Total area of wing
NC, NB	=	Chordwise and spanwise number of panel segments of wing
NX, NT	=	Axial and azimuthal panel segments of body
A	=	Area of panel, body or wing
dz/dx	=	Panel slope
α	=	Angle of attack
$\bar{c}$	=	Moment arm about reference point.

2.2 SURFACE SPLINE FIT

A versatile surface spline fit method (Reference 6) has been incorporated into the aero code for the following purposes. Firstly, from the experience gained in the operation of the code developed for the analysis of missile wings, it is clear that input preparation must be greatly simplified for any wing surface except for the plane one. The calculation of wing camber,

thickness and twist at the locations as specified in the aero code (control point of each panel) proved to be difficult and time consuming. Thus, a surface spline fit that is efficient and flexible is strongly suggested. Secondly, the pressure loads on the missile as computed by the code are originally prescribed at the control point of aero panels. To satisfy the requirement of structural analysis the pressure loads at the centroid of each finite element are needed. Interpolation of the pressure loads therefore becomes necessary.

2.2.1 Theoretical Discussion

The one-dimensional spline curve is the numerical analogue to the french curve used in curve plotting. It is based on the premise that by bending a thin flexible beam that is constrained to pass through a number of coordinates, a smooth curve is formed. Mathematically, the curve represented by the spline is the thin-beam equation whose solution is the desired spline.

Analogous to the one-dimensional spline, the surface spline is that of a thin plate subject to a bending moment while it is constrained at a number of points. The differential equation for the plate is

$$\nabla^4 \delta = P \tag{19}$$

where

δ and P represent the deflection and load, respectively.

The solution that satisfies the equilibrium conditions and vanishingly small deflection at large distance is

$$\delta(x_j, y_j) = k_1 + k_2 x_j + k_3 y_j + \sum_{i=1}^{N} k_{i+3} r^2_{ij} \ln\left(r^2_{ij}\right) \qquad j = 1, N \tag{20}$$

In Equation 20, $r^2_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2$ and k_1, k_2, and k_3, and k_i are coefficients describing the deflection of the plate. These coefficients, which total 3 + N, are determined using the constraints at N points, i.e., Equation 20 and the equilibrium conditions

$$\sum_{i=1}^{N} K_i = 0, \quad \sum_{i=1}^{N} K_i x_i = 0, \quad \sum_{i=1}^{N} K_i y_i = 0 \tag{21}$$

These yield the necessary and sufficient conditions for their evaluation.

The calculation of the deflections, δ, at any arbitrary location (x, y) is achieved by substituting the calculated values of coefficients and computing from Equation 20

$$\delta(x, y) = k_1 + k_2 x + k_3 y + \sum_{i=1}^{N} K_{i+3} r_i^2 \ln\left(r_i^2\right) \tag{22}$$

where

$$r_i^2 = (x-x_i)^2 + (y-y_i)^2$$

2.2.2 Application

In interpolating either the wing surface geometry or the aerodynamic load over wing surfaces, it is necessary to first calculate the 3 + N coefficients using the wing description data. The desired value of the interpolated quantities at any location can then be evaluated using Equation 22.

It will be appropriate to mention some advantages of this particular spline method as compared to others. Generally, spline fit methods require large storage for the coefficients. In the two-dimensional case, the storage requirement is approximately 16N, where N is the number of data points given. For this method of surface fit only (N + 3) storage locations are required. In addition, searching for the correct patch on which the desired point of interpolation is located is avoided thereby increasing computing efficiency. Further, the method is capable of accepting input data given at points randomly located. Hence, additional processing is eliminated in these cases. Especially important is the fact that some locations of the finite element centroid can and often do lie outside the domain of the control point of the panel. In general most methods fail when it is required to interpolate at such locations. The present method, however, can interpolate at these locations without special provisions.

Section 3
COMPUTER PROGRAM

The present version of the MDAC aerodynamic code is developed to provide a practical and user-oriented tool for the purpose of analyzing the pressure load on a missile or its components. Using the output of NSWC computer code PING or BING which specifies the finite element and grid point distribution on the missile, it computes the aerodynamic pressures on the missile and interpolates by surface fit to produce the desired pressure load at the centroid of each element. Punch card output in the bulk data format of NASTRAN are generated as desired for subsequent structural analysis.

In the following sections, the capabilities and restrictions of the present code will be discussed, followed by the input description for all three test cases. Outputs at the end of computation are reproduced to aid users.

3.1 CAPABILITIES OF THE MDAC AERO CODE

As presently modified, the code is capable of computing the pressure load on a missile, or if desired its component such as wing, body or tail, for both subsonic and low-supersonic flows. Restricted to small disturbances, the code analyzes a slender body of circular or near circular cross-section at zero or small angle of attack. Wings of various forms are represented as the sum of camber, thickness and twist slopes. The pressure load at the control point of each panel whether it lies on the body or on the wing are computed from the strength of the distributed sources satisfying the tangency condition at the panel surface. The pressure load at the centroid of the finite element as generated from NSWC codes is next computed by interpolation using the surface fit method successfully tested in the earlier effort. They are produced as punched card output in the bulk data format of NASTRAN.

For use on the CDC CYBER 74 computer with KRONOS 2.1 system, the code requires a field length of 160,000 using the FTN compiler. As written, it

permits the use of 700 finite elements and 350 grid points for the body and wing each. For the case where the number of finite elements or grid points exceeds the above limit, updating of the program will be necessary.

The code has certain limitations when treating a missile with complex configuration that are inherent in the original MDAC-Woodward program. One of them is the missile attitude restrictions. Although it can be removed by modification of the code, at present the code can only be used to treat the case with small angle of attack without any out of pitch-plane attitude changes (roll or yaw). The code assumes that the missile configuration is composed of one main fuselage, multiple wings and two pairs of nacelles. Consequently, a multiple body problem cannot be analyzed without code alterations. Furthermore, the cross section of the body must be approximately circular for the calculation to be valid. These restrictions, however, are not fundamental to the basic theory of small disturbances and hence can be dealt with successfully in future development.

3.2 PROGRAM INPUT DESCRIPTION

Numerous input parameters are required in the present code for correct numerical computation of a desired case. To minimize the labor of input preparation, their numbers have been kept to a minimum and appropriate default values for some parameters depending on the case analyzed have been set in the code.

The input parameters and data have been organized into two main groups in accordance with their assigned functions similar to the overall organization of the code. The first group contains all the parameters needed for the specification of case selection, missile configuration and flow conditions. They form the input parameter of the namelist UNIFID. The second group specifies the finite element and grid point system as generated by the NSWC computer codes BING and/or PING. It includes the specification for the coordinate system used for the finite elements.

For the convenience of the user, card format for inputs are described in their sequential order as required by the code.

(A) Aerodynamic Control Parameters

Card 1 Title and case description, format (20A4), alphanumeric

Card 2 $UNIFID, begins at Column 2 with the symbol $.

Card 3 and additional cards if necessary, contain the following parameters as input.

ICASE = 1	Wing only
= 2	Body only
= 3	Complete missile
NTRANS	Number of coordinate transformations required to convert the coordinates for the finite element to the coordinates used in the aerodynamic code. Two transformations are allowed for each missile component, in the following order; body, wings and nacelles.
IPUNCH = 0	No punched card output, default value
= 1	Punched card output requested
IRW = 0	Computation results are not saved on tape, default value
= 1	Aerodynamic computation results are saved on tape 12 for restart, first run only.
= 2	Restart run, bypass all aerodynamic computations and begins computation at the start of pressure interpolation.
POLAR	Number of incremental angles of attack, default value = 0
XMACH	Mach number of flow, default value = 2.0
PINF	Free stream pressure, default value = 14.7 psi
DADEG	Incremental angle of attack of missile in degrees, default value = 0.
SID	Case identification number for each missile component (up to 20 allowed), default values have been set at 1. Also used for selecting the sign convention for pressure output. Code uses the accepted rule for pressure, i.e. positive acting toward the surface and negative acting away from the surface. For opposite sign convention, set SID(L) as negative for pressure output on

the desired component L (body, wing surfaces, upper or lower).

The following input parameters are required for the case where missile body is included.

NBPFL	Number of body profile stations for specifying body geometry $r = r(x)$, up to 51 stations allowed
XB $\leq$ 51	Axial coordinate of body stations in the direction of flow for specifying the body profile.
RB $\leq$ 51	Radial coordinate of body profile at corresponding body stations
ZD $\leq$ 51	Body camber at corresponding body stations
ARB	Angle of attack of missile body in degrees, default value = 0

The following inputs are required for the case where the missile wing is included.

ISØLID = 0	Wing of built-up construction, pressure load on both upper and lower wing surface generated, default value
= 1	Wing of solid section, sum of the pressure loads on wing surface generated
IFØRM = 0	Wing of built-up construction, both upper and lower surface height (ZWI) are given at identical locations (XWI, YWI)
= 1	Wing of built-up construction, upper and lower surface height (ZWI) are not given at identical locations
= 2	Flat wing surface
NWPI	Number of coordinate points specify the wing surface contour ($\leq$30). Code allows a maximum of 20 wing surfaces (upper and lower). Each wing surface must be defined in the form of equation of a surface, ZWI = f(XWI, YWI). A maximum of 30 points are permitted with the provision that the first four points are restricted to the specification of the corners of the wing only. The coordinates for the corners must be given in the following order. Starting with the innermost

	point of the leading edge, the remaining corners are to be given in a clockwise order viewing from the above.
NWING	Number of wings or tails. A maximum of 10 is allowed, default value = 0.
XWI(i, j), YWI(i, j), ZWI(i, j)	Wing surface coordinates where indices i and j designate the surface coordinate point and wing surface, respectively.
DIHED	Dihedral angle, degrees
PIVØT	Indicates dihedral, 0 for no dihedral and >0 for dihedral present
ARW	Angle of attack of wing in degrees, with respect to the body axis if complete missile is considered, default value = 0

The following input parameters are required for the case where nacelle is present.

	XNACEL	Number of nacelles, default value = 0
	NACELP	Number of stations defining the nacelle profile, restricted to a maximum of 51 points each
	XN ≤ 51	Axial coordinate of nacelle stations in the direction of flow for specifying the nacelle profile
	RN ≤ 51	Radial coordinate of nacelle profile at corresponding nacelle stations
	ZN ≤ 51	Nacelle camber given at the corresponding nacelle station
	ARN	Angle of attack of nacelle in degrees with respect to the body axis, default values have been set at 0.0.
Card 4	$END	Begin at column 2 the symbol $ for ending input.

(B) Finite Element and Grid Inputs

The second group of input cards are arranged immediately following Card 4 as follows:

Cards 5 and 6 Coordinate system specifications in the format of NASTRAN bulk data deck [page 2.4-49 - 2.4-54, NASTRAN User's Manual, NAS SP-221(01), 1972]. Three position vectors, A, B and C, are used to define the coordinate system. The

first defines the origin. The second defines the Z axis and the third defines a point in the XZ plane.

For Card 5 the following format is used:

Col 1-8, Coordinate system CØRDJRbb for rectangular, CØRDJCbb for cylindrical and CØRDJSbb for spherical where J identifies the coordinate system numbers.

Col 9-16, Coordinate identification number J, integer

Col 17-24, Reference coordinate system, integer, optional

Col 25-32, 33-40, 41-48, Components of vector A (3F8.2)

Col 49-56, 57-64, 65-72, Components of vector B (3F8.2)

For Card 6: Col 1-8, blank

Col 9-16, 17-24, 25-32, Components of vector C (3F8.2)

Cards 5 and 6 are to be repeated as many times as the number of coordinate systems (NTRANS) required for specifying the finite element. They are to be arranged in the order of body, wings and nacelles if any.

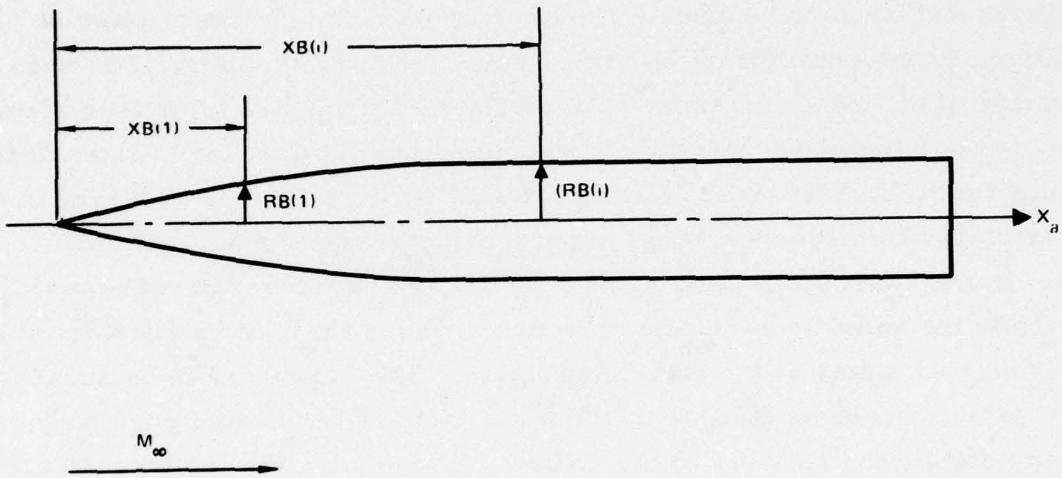
The finite element and grid specifications are arranged immediately following the coordinate system input in the same order, i.e., body, wings and nacelles.

Card 7 Specifies the number of grid cards (NGRIDP) and element cards (NE), Format (2I10).

The grid and the element cards as generated by the NSWC computer code BING are first sorted into two separate groups accordingly. They are placed immediately after card 7 with the grid cards in front.

Some remarks in regard to the input preparation need be made here to facilitate the use of the aero code.

For the body-only case, inputs of the body profiles, expressed as $r = r(x)$, for aero panel generation must be in the aero coordinate system. It is required that x be the axial distance from the tip and r the radius of the body cross section (Figure 3). For the finite element and grid point, their rela-



X_a - AXIS IN THE FLOW DIRECTION
Y_a - AXIS TO THE RIGHT
Z_a - AXIS POSITIVE UPWARD

Figure 3. Body Profile Input

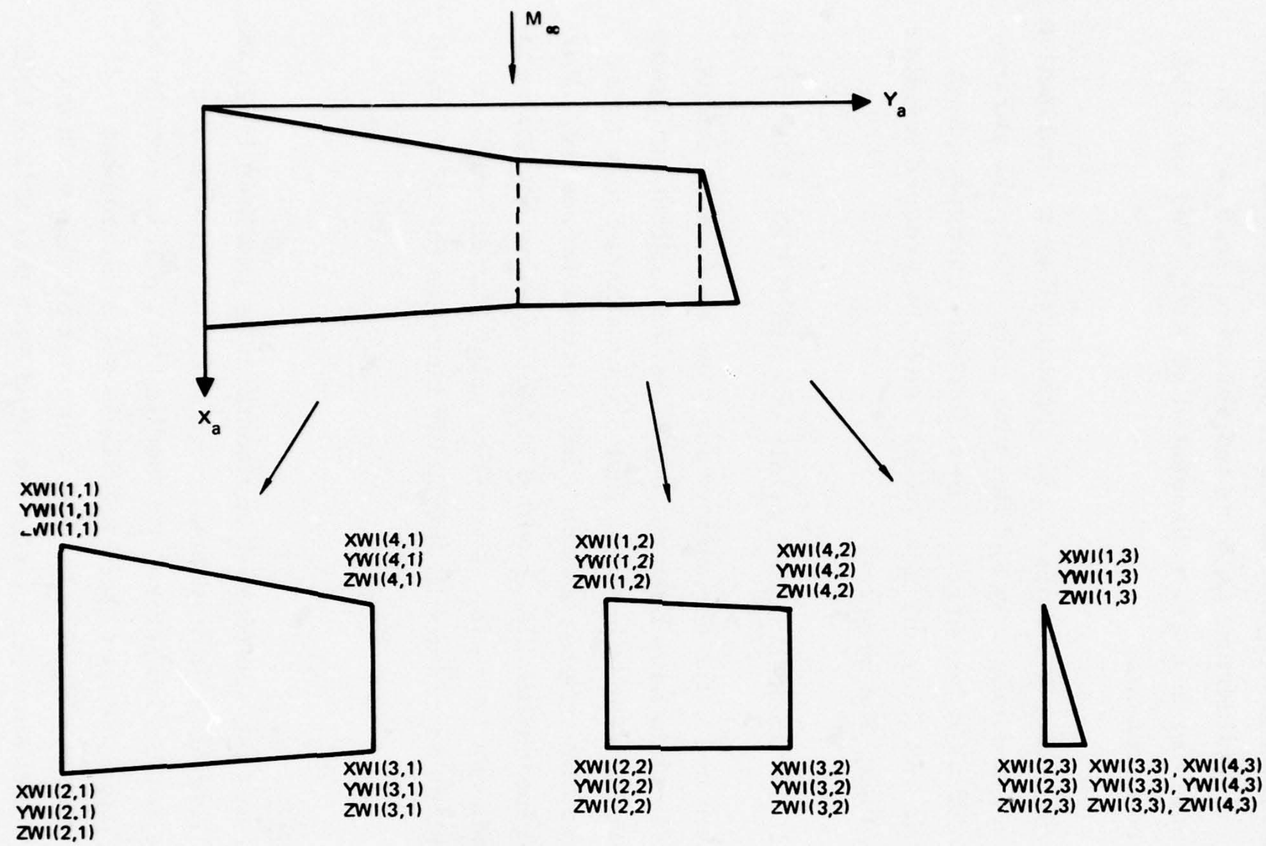
tionship to the basic coordinate system and the basic coordinate system to aero coordinate system are to be specified in the coordinate input card immediately following the namelist input. The grid card and the element card are to be separated and placed after the coordinate system specification with a card specifying their numbers placed behind the coordinate card.

For the wing only case, its input requires the specification of both the wing plan form and its surface contour. The input used for the description of wing planform are the coordinates of corner points and break points. For each individual wing, inputs for those of quadrilateral shape are their four corner coordinates. The codes as written have allocated the first four values of the coordinates XWI, YWI and ZWI for each wing or wing region, arranged in a counterclockwise order starting from the leading-edge corner or break point. In the case of triangular wings, there are only three corner points. To satisfy the same requirement, the apex or wing tip is to be considered as two coincident points of the same coordinate. The inputs are to be arranged in the same manner as that of quadrilateral shape with leading edge body-corner point as the starting point. Wings with complex shapes (see Figure 4) must be subdivided into appropriate quadrilateral or triangular regions and inputs prepared accordingly. The only requirement is the condition that the control chords, apart from either the leading edge or the trailing edge, must be in the streamwise direction.

All inputs for the wing surface are to be given in the form of equation of surface $Z = f(x, y)$ in terms of XWI, YWI and ZWI. Starting with the fifth value, XWI(5, I), YWI(5, I), ZWI(5, I), their total number may not exceed 26 points. For wings with flat surfaces, only the first four points are necessary.

Inputs defining wing thickness, camber and twist have been eliminated to reduce the input preparation effort. The code generates the required values of camber, thickness slope and twist angle for the wing using the existing surface fit scheme from the input.

The finite element and grid point coordinate system cards are placed similar to the body only case immediately behind the namelist input. They are followed by a card specifying the number of grid cards and element cards. The



NWING = 3, NTRANS (1) = 1, 1, 1,
IF BUILT-UP WING IS USED,NWPI(1), NWPI(2), AND NWPI(3) MUST BE SPECIFIED

Figure 4. Subdivision of Complex Wing

grid point and element card as generated by PING occupies the last position in the input.

Coordinate system used for the wing can be either one of the three systems, i. e., rectilinear (x, y, z), cylindrical (r, θ, z) and spherical (r, θ, ϕ). In terms of the input parameters they are represented by XWI, YWI and ZWI, respectively for all three systems.

For the complete missile case, the input can be considered as a combination of these two cases, i. e., the body only and the wing only. Only the parameters ICASE and NTRANS need be corrected to reflect the changes. Both coordinate system cards and finite element cards are to be grouped together in the order of body, wings, and nacelles.

3.3 VARIOUS RESTRICTIONS AND CODE CHANGES REQUIRED FOR THEIR REMOVAL

In developing the present code, the dimensions for wing segments, panels, finite elements and grid points have been set based on the estimate of future requirements. Nevertheless, it is possible that circumstances may arise when one or more of the dimensions, as specified, need be increased. For convenience, pertinent information is provided in the following section necesary changes in the code can be made. It must be noted that any further increase in the dimensions as coded will inevitably increase the field length and CPU time.

3.3.1 Input Parameters

The maximum number of wing sections or horizontal tails (NWING) is limited to a total of 10. This limitation is imposed by the original aerodynamic analysis section of the code. Therefore, its modification will be very involved and costly. The wing panel number (XNB x XNC) is set in subroutine DEFAUT as 6 x 11 or 66 for NWING = 1. For a number of wing sections greater than one, the panel number must be modified such that XNB x XNC x NWING $\leq$ 110. This can best be achieved by changing the value of the wing panel number through the selecting of appropriate values of XNB and XNC.

3.3.2 Finite Element and Grid Point

The maximum number of finite elements and grid points have been set at 700 and 350, respectively, for body, wing and horizontal tail. Their modification can be made by changing the dimensions of variables XC, NG, NELM in common statement PBDATA as it appears in subroutines PBODY, PSUM, WINGPR, WINGO, SAVE and PNACEL.

Correspondingly, changes are to be made also to the dimension statement of variables XI and PB in subroutine PBODY, variable XI in subroutine PWING, variables XI and PB in subroutine PNACEL.

Section 4
TEST CASES

A test case was selected for aerodynamic analysis by the present code for the purpose of validating the theoretical approach and code verification. A missile of swept wing with ogive-cylinder body as described in Reference 8 was chosen for the purposes. Its geometry and panels are shown in Figure 5 and the wing cross section is shown in Figure 6. In the noted reference experimental measurements were made for flow at Mach number 2.01 with the model placed at different angles of attack and sideslips.

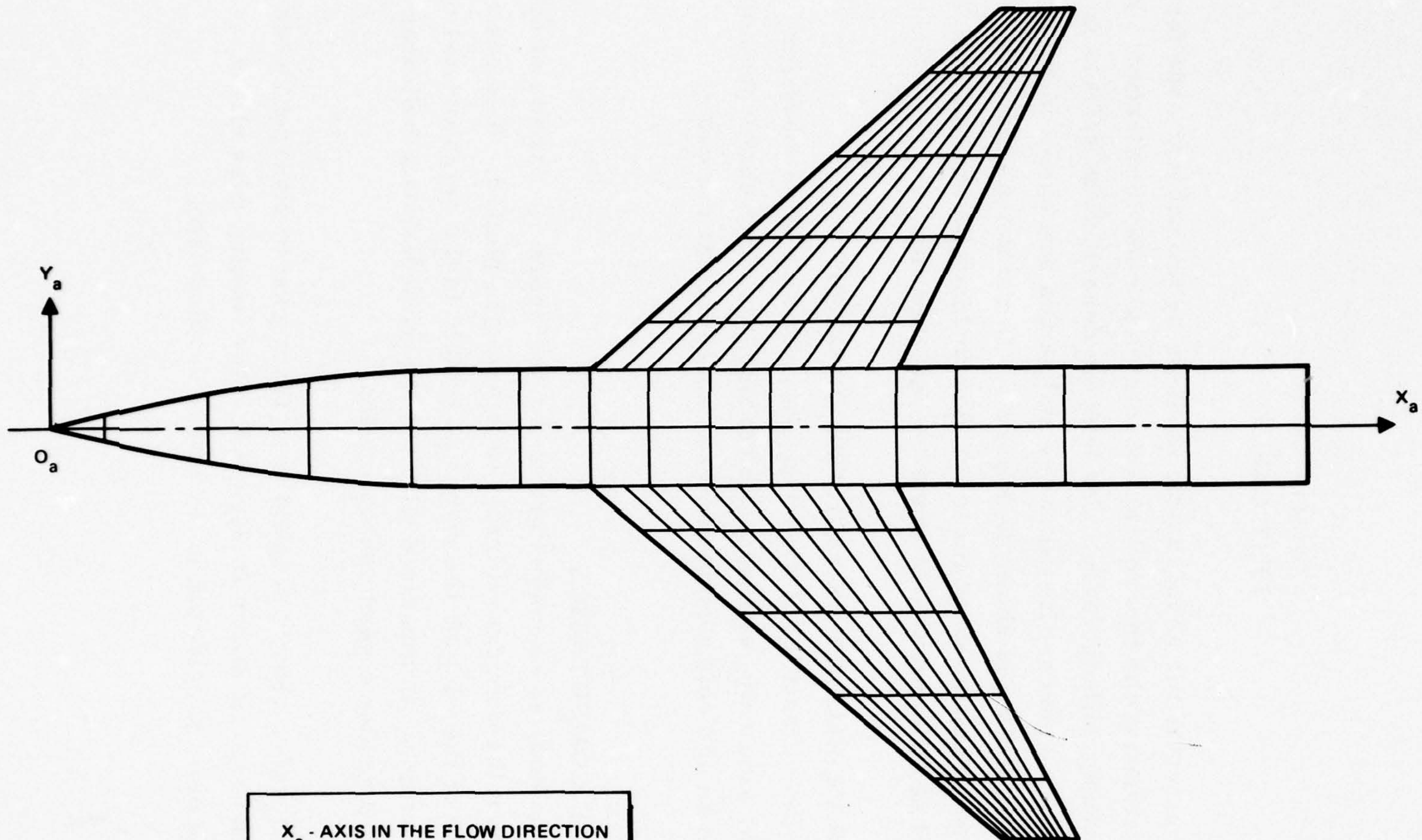
4.1 INPUT ARRANGEMENTS AND OUTPUT PRINTOUT

To demonstrate the versatility of the code, computations were made for (a) body only, (b) wing only, and (c) complete missile. The corresponding input arrangements and output printouts are reproduced at the end of Section 4.

4.2 RESULTS OF COMPUTATION

Pressure loads on body as computed are shown in Figure 7. Effects of the wing obtained from the complete missile case are also plotted. It is seen that the influence of the wing on the body lies mainly in the neighborhood of the wing-body junction. Comparison with the experimental data indicates that the analysis provides a good approximation.

For the wing only case, computed loads vary from good to divergent when compared with the data as shown in Figure 8. The leading edge effect apparently needs to be considered in the future development.



X_a - AXIS IN THE FLOW DIRECTION
Y_a - AXIS TO THE RIGHT
Z_a - AXIS POSITIVE UPWARD

Figure 5. Test Case Wing Body Combination

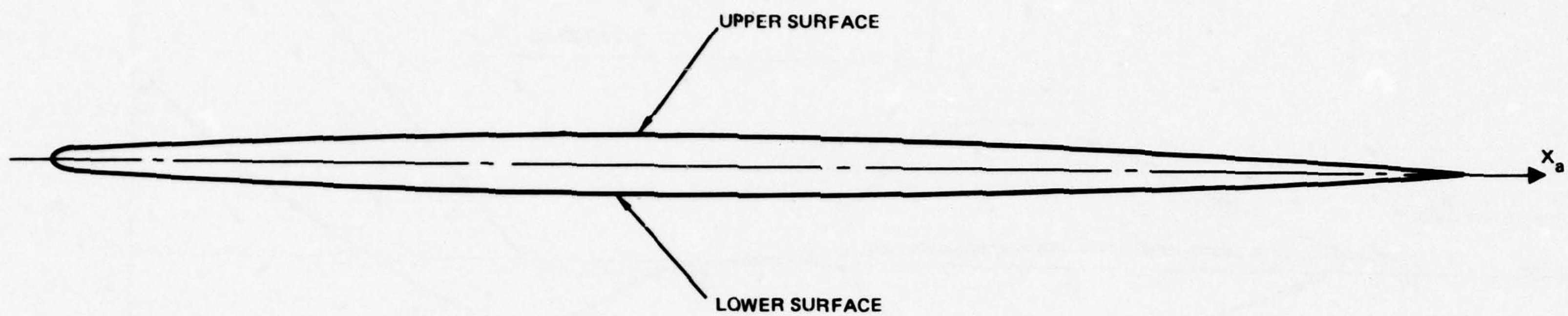


Figure 6. Wing Section - NACA 64A005

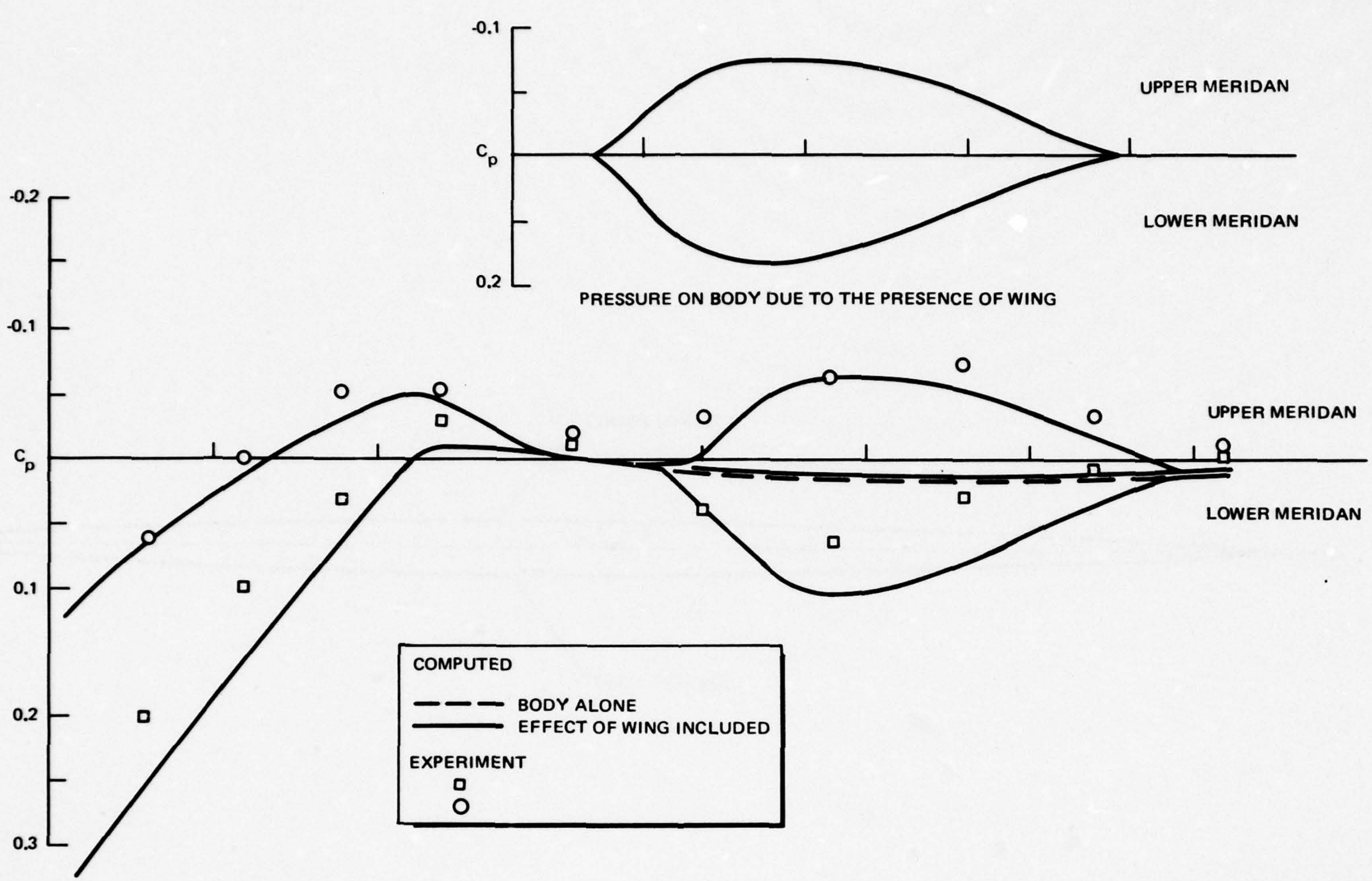


Figure 7. Pressure Distribution on Body at Mach 2.01 and 5° Angle of Attack

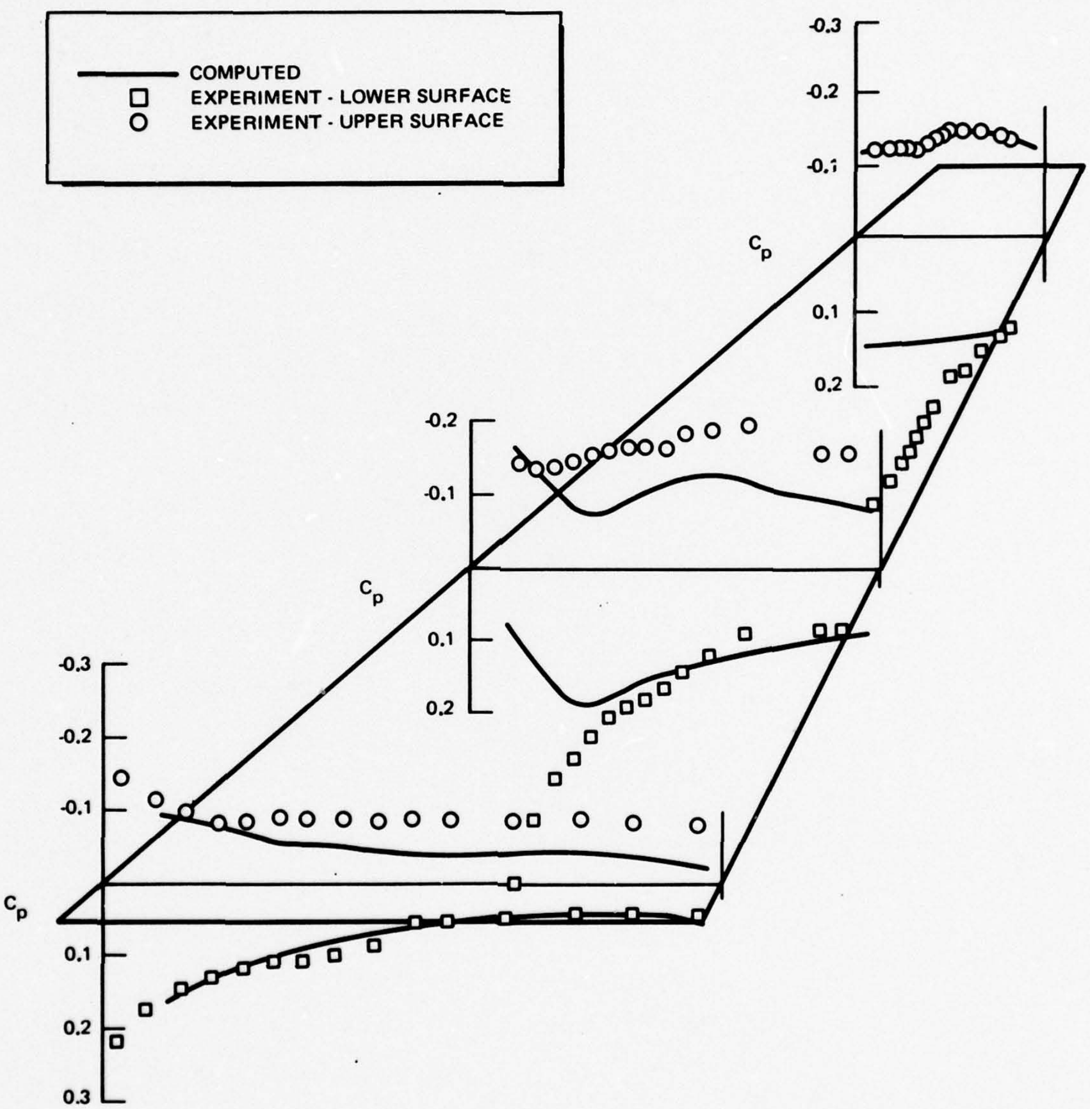


Figure 8. Pressure Distribution on Wing at Mach 2.01 and 5° Angle of Attack

4.3 INPUT CARDS FOR TEST CASES

4.3.1 Body Only Case

```
   FINAL TEST, BODY ALONE, NSWC CARD INPUT
 $UNIFID
   ICASE = 2, NTRANS(1) =1, N3PFL = 24, XMACH = 2.01, ARB = 5.0,
  XB(1) = 0.0, 0.5833, 1.1667, 1.75, 2.333, 2.916,  3.5,  4.0833, 4.6667,
  5.25,  5.8333, 6.4167,  7.0,  7.5833, 8.1667, 8.75, 9.333, 9.9167,
  10.5,  11.0833, 11.6667, 15.2,  25.0,  36.5,
   RB(1) = 0.0,  0.1655, 0.3333, 0.4694, 0.6079, 0.7376, 0.8585, 0.9708,
  1.0745, 1.1696,  1.2563, 1.3346, 1.4045, 1.4661,  1.5194,  1.5645,
  1.6013, 1.6299, 1.6503, 1.6626, 1.6667, 1.6667, 1.6667, 1.6667,
 $END
CORD1C          1              0.0       0.0       0.0     1.0      0.0       0.0
                1.      0.     1.0
      15             5
GRID           45        1   1.351 210.000    7.796        1        4
GRID           46        1   1.351 240.000    7.796        1        4
GRID           58        1   1.485 240.000    9.728        1        4
GRID           43        1   1.351 150.000    7.796        1        4
GRID           55        1   1.485 150.000    9.728        1        4
GRID           56        1   1.485 180.000    9.728        1        4
GRID           40        1   1.351  60.000    7.796        1        4
GRID           41        1   1.351  90.000    7.796        1        4
GRID           53        1   1.485  90.000    9.728        1        4
GRID           48        1   1.351 300.000    7.796        1        4
GRID           60        1   1.485 300.000    9.728        1        4
GRID           61        1   1.485 330.000    9.728        1        4
GRID           50        1   1.485   0.000    9.728        1        4
GRID           51        1   1.485  30.000    9.728        1        4
GRID           63        1   1.530  30.000   11.660        1        4
CTRIA2         90       90      40       41       53    0.000
CTRIA2         95       95      56       55       43    0.000
CTRIA2        100      100      45       46       58    0.000
CTRIA2        105      105      61       60       48    0.000
CTRIA2        110      110      50       51       63    0.000
```

The input cards as shown, in sequence, are:

(I) Title Card

Notes: For input of any descriptive title

Format: (20A4)

(II) Namelist UNIFID input

Notes: For input of aerodynamic control parameters appropriate to the case of interest.

Format: In accordance with the input variables, real or integer.

Test Case:

ICASE = 2, body only case

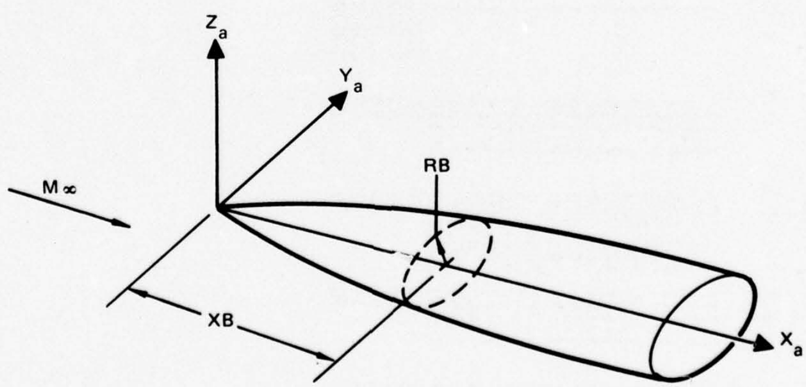
NTRANS(1) = 1, no intermediate coordinate system

XMACH = 2.01, flow Mach number

NBPFL = 24, number of point for body profile input

ARB = 5.0, angle of attack of body

For body profile inputs they are the radius RB and axial distance XB using the cylindrical coordinate system oriented as shown in the sketch. X_a, Y_a and Z_a represent the aerodynamic coordinate system.



(III) Coordinate Card

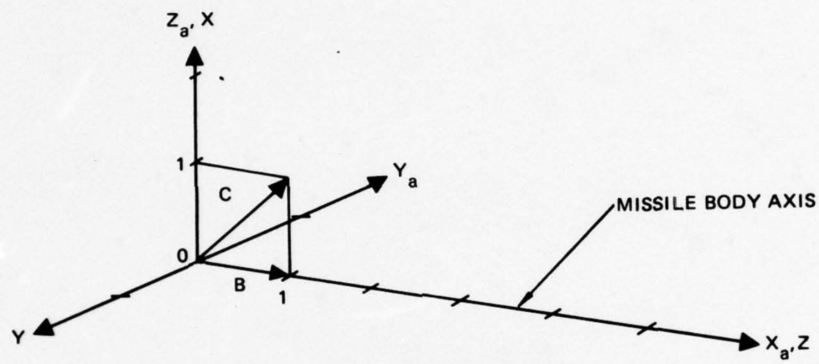
Notes: The coordinate cards are needed for input of vectors A, B and C used in defining the relations between the coordinate systems of the finite element (X, Y, Z) and the aero code (X_a, Y_a, Z_a).

In accordance with NASTRAN, vector A defines the origin, sector B defines the Z axis and vector C defines the X-Z plane.

Format: NASTRAN bulk data deck, first card (A3, 2I8, 6F8.2) and second card (A8, 3F8.2).

Test Case:

Coordinate systems for the finite element and the aero codes are shown in the sketch.



Since their origin coincides, vector A becomes a null vector or (0.0, 0.0, 0.0). With the Z axis coincident with the X_a axis, vector B has components (1.0, 0.0, 0.0). As vector B lies in the $X_a Z_a$ plane, its components are (1.0, 0.0, 1.0).

(IV) Finite Element Cards

Notes: The finite element and grid point cards as generated by BING or PING are used for defining element coordinates. They must be divided into two separate groups with the grid point card preceding the element card. A card specifies that the numbers of grid card and element card in a format 2I10 must be placed before them.

Format: For grid and element number card (2I10), grid and finite element cards must be the same as generated by BING or PING.

Test Case: As shown.

4.3.2 Wing Only Case

```
  FINAL TEST CASE,  WING ONLY,  NASA MEMO 10 - 15 - 58L
$UNIFID
    ICASE = 1, NTRANS(1) = 1, NWING = 1, NWPI(1) = 30, 30, IFORM(1) = 1,
   ARW = 5.0,
  XMACH = 2.01,
   XWI(1,1) = -1.665, -1.665,  -11.995, -11.995,
 XWI(5,1) = -1.665, -1.665, -1.665, -3.731, -3.731, -3.731, -3.731,
 -3.731, -5.797, -5.797, -5.797, -5.797, -5.797, -7.863, -7.863, -7.863,
 -7.863, -7.863, -9.929, -9.929, -9.929, -9.929, -9.929, -11.995,
  -11.995, -11.995,
  YWI(1,1) = 0.0,  8.89,  14.05,  12.05,
  YWI(5,1) = 2.2225, 4.4450, 6.6675, 2.410, 4.288, 6.1661, 8.0441, 9.922,
 4.820, 6.3535, 7.887,  9.4204, 10.954, 7.230, 8.419, 9.608, 10.797,
 11.986, 9.640, 10.4844, 11.329, 12.1735, 13.018, 12.55, 13.05, 13.55,
 ZWI(5,1) = 0.1594, 0.1731, 0.1044,0.00, 0.1347, 0.1463, 0.0882, 0.0007,
 0.00, 0.11, 0.1195, 0.072, 0.0006, 0.00, 0.0852,  0.0926, 0.0558,
 0.0004,  0.00, 0.0605, 0.0658, 0.0397, 0.0003,  0.0358, 0.0389, 0.0235,
   XWI(1,2) = -1.665, -1.665,  -11.995, -11.995,
 XWI(5,2) = -1.765, -1.765, -1.765, -3.831, -3.831, -3.831, -3.831,
 -3.831, -5.897, -5.897, -5.897, -5.897,-5.897, -7.963, -7.963,-7.963,
 -7.963, -7.963, -9.999, -9.999, -9.999, -9.999,  -9.999,  -11.895,
 -11.895, -11.895,
  YWI(1,2) = 0.0,  8.89,  14.05,  12.05,
 YWI(5,2) = 2.1225, 4.345, 6.567, 2.510, 4.388, 6.266, 8.144, 9.94,
 4.92, 6.43, 7.96, 9.45, 11.0, 7.33, 8.519, 9.708, 10.897,
 11.996, 9.76, 10.58, 11.42, 12.27, 13.118, 12.55, 13.15, 13.55,
 ZWI(5,2) =-0.1594,-0.1731,-0.1044,0.00,-0.1347,-0.1463,-0.0882,-0.0007,
 0.00,-0.11,-0.1195,-0.072,-0.0006,-0.00,-0.0852, -0.0926,-0.0558,
-0.0004,  0.00,-0.0605,-0.0658,-0.0397,-0.0003, -0.0358,-0.0389,-0.0235,
$END
CORD2R        2        15.59    0.0      0.0      15.59    0.0      1.0
        15.59    -1.0     1.0
CORD2R        2        15.59    0.0      0.0      15.59    0.0      1.0
        15.59    -1.0     1.0
       24       10
GRID        251       2  1.6650  -.0000 0.0000        2
GRID        252       2  1.6650  2.2225  .1594        2
GRID        261       2  3.7310  2.4100 0.0000        2
GRID        262       2  3.7310  4.2880  .1347        2
GRID        271       2  5.7970  4.8200 0.0000        2
GRID        272       2  5.7970  6.3535  .1100        2
GRID        281       2  7.8630  7.2300  .0000        2
GRID        282       2  7.8630  8.4190  .0852        2
GRID        291       2  9.9290  9.6400  .0000        2
GRID        292       2  9.9290 10.4844  .0605        2
GRID        301       2 11.9950 12.0500  .0000        2
GRID        302       2 11.9950 12.5500  .0358        2
GRID        254       2  1.6650  6.6675  .1044        2
GRID        255       2  1.6650  8.8900  .0000        2
GRID        264       2  3.7310  8.0441  .0882        2
GRID        265       2  3.7310  9.9220  .0007        2
GRID        274       2  5.7969  9.4204  .0720        2
GRID        275       2  5.7970 10.9540  .0006        2
GRID        284       2  7.8630 10.7970  .0558        2
GRID        285       2  7.8630 11.9860  .0004        2
GRID        294       2  9.9290 12.1735  .0397        2
GRID        295       2  9.9290 13.0180  .0003        2
GRID        304       2 11.9950 13.5500  .0235        2
GRID        305       2 11.9950 14.0500  .0002        2
CQUAD2      500     500     251     261     262     252 0.0000
CQUAD2      501     501     261     271     272     262 0.0000
CQUAD2      502     502     271     281     282     272 0.0000
CQUAD2      503     503     281     291     292     282 0.0000
CQUAD2      504     504     291     301     302     292 0.0000
CQUAD2      515     515     254     264     265     255 0.0000
```

CQUAD2	516	516	264	274	275	265	0.0000
CQUAD2	517	517	274	284	285	275	0.0000
CQUAD2	518	518	284	294	295	285	0.0000
CQUAD2	519	519	294	304	305	295	0.0000
	24	10					
GRID	251	2	1.6650	-.0000	0.0000	2	
GRID	312	2	1.6650	2.2225	-.1594	2	
GRID	321	2	3.7310	2.4100	0.0000	2	
GRID	322	2	3.7310	4.2800	-.1347	2	
GRID	331	2	5.7970	4.8200	0.0000	2	
GRID	332	2	5.7970	6.3535	-.1100	2	
GRID	341	2	7.8630	7.2300	-.0000	2	
GRID	342	2	7.8630	8.4190	-.0352	2	
GRID	351	2	9.9290	9.6400	-.0000	2	
GRID	352	2	9.9290	10.4844	-.0605	2	
GRID	361	2	11.9950	12.0500	-.0000	2	
GRID	362	2	11.9950	12.5500	-.0358	2	
GRID	314	2	1.6650	6.6375	-.1064	2	
GRID	255	2	1.6650	8.0900	.0000	2	
GRID	324	2	3.7310	8.0441	-.0332	2	
GRID	325	2	3.7310	9.9220	-.0007	2	
GRID	334	2	5.7969	9.4204	-.0720	2	
GRID	335	2	5.7970	10.9540	-.0006	2	
GRID	344	2	7.8630	10.7970	-.0558	2	
GRID	345	2	7.8630	11.9860	-.0004	2	
GRID	354	2	9.9290	12.1735	-.0397	2	
GRID	355	2	9.9290	13.0180	-.0003	2	
GRID	364	2	11.9950	13.5500	-.0235	2	
GRID	365	2	11.9950	14.0500	-.0002	2	
CQUAD2	525	525	251	321	322	312	0.0000
CQUAD2	526	526	321	331	332	322	0.0000
CQUAD2	527	527	331	341	342	332	0.0000
CQUAD2	528	528	341	351	352	342	0.0000
CQUAD2	529	529	351	361	362	352	0.0000
CQUAD2	540	540	314	324	325	255	0.0000
CQUAD2	541	541	324	334	335	325	0.0000
CQUAD2	542	542	334	344	345	335	0.0000
CQUAD2	543	543	344	354	355	345	0.0000
CQUAD2	544	544	354	364	365	355	0.0000

The input cards as shown, in sequence, are:

(I) Title Card

Notes: For input of any descriptive title

Format: (20A4)

(II) Namelist UNIFID input

Notes: For input of aerodynamic control parameters appropriate to the case.

Format: In accordance with the input variables whether real or integer.

Test Case:

ICASE = 1, wing only case

NTRANS(1) = 1, no intermediate coordinate system

NWING = 1, one wing section

NWPI(1) = 30, (2) = 30, number of points of input defining the upper and lower surface contours given at different locations of the upper and lower surfaces.

IFORM(1) = 1, signifies the fact that the wing surfaces are of airfoil type for the present case.

XMACH = 2.01, flow Mach number

ARW = 5.0, wing angle of attack

XWI(i), YWZ(i) and ZWI(i) represent the wing surface Z = f(x, y), with ZWI as the height of wing surface. They can be given in either rectangular, cylindrical or spherical coordinate systems as desired. The first four points must be used to specify the wing planform as shown in the sketch. Therefore, for the test case, we have

XWI(1, 1) = -1.665, -1.665, -11.995, -11.995

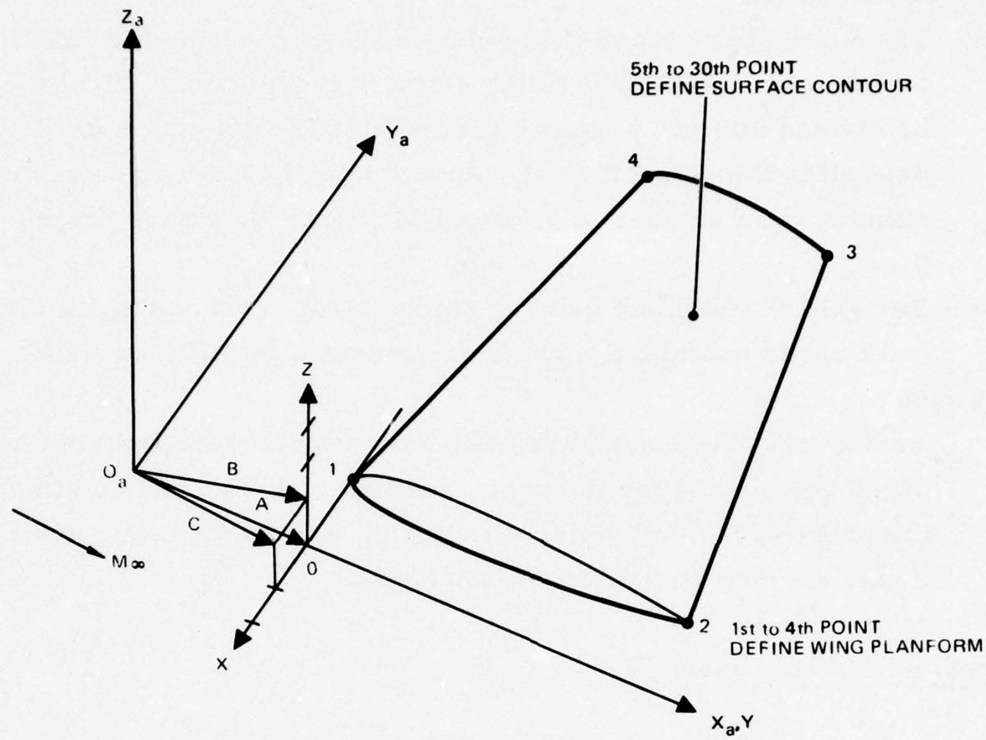
XWI(5, 1) = -1.665, -1.665, ... ,

YWI(1, 1) = 0.0, 8.89, 14.05, 12.05,

YWI(5, 1) = 2.2225, 4.445, 6.6675, ...

ZWI(5, 1) = 0.1594, 0.1731, 0.1044, ... etc.

It should be noted that all the inputs for wing surface coordinates must be that of the right wing of the missile. The coordinate system for the wing (X, Y, Z) and the aero code (X_a, Y_a, Z_a) are shown in the sketch.



(III) Coordinate Card

Notes: The coordinate cards are read for input of vectors A, B and C used in defining the relation between the coordinate systems of the finite element (X, Y, Z) and aero code (X_a, Y_a, Z_a). In accordance with NASTRAN, vector A defines the origin, vector B defines the Z axis and vector C defines the X-Z plane.

Format: NASTRAN bulk data deck, first card (A8, 2I8, 6F8.2) and second card (A8, 3F8.2).

Test Case:

As shown in the previous sketch, the vector A has components (15.59, 0.0, 0.0), vector B (15.59, 0.0, 1.0) and vector C (15.59, 1.0, 1.0). Two sets of coordinate cards are needed for the case as the upper and lower surfaces are specified separately at different locations.

(IV) Finite Element Cards

Notes: The finite element and grid point cards as generated by BING or PING are used for defining element coordinates. They must be divided into two separate groups with the grid point card preceding the element card. A card specifies the grid card and element card number in a format 2I10 must be placed before them.

Format: For grid and element number cards (2I10), grid and finite element cards must be the same as generated by BING or PING.

Test Case:

As shown in the input listing two sets of grid point and element cards are needed for the upper and lower surfaces. To simplify the presentation and reduce computing effort only a few elements are selected for pressure outputs.

4.3.3 Complete Missile Case

```
   FINAL TEST CASE,  COMPLETE MISSILE,  NASA MEMO 10 - 15 - 58L
 $UNIFID
  ICASE=3, NTRANS(1)= 1, 1, NBPFL= 24, NWPI(1) = 30, NWING = 1,
   ARB = 5.0,  XMACH = 2.01,
  XB(1) = 0.0, 0.5833, 1.1667, 1.75, 2.333, 2.916,  3.5,  4.0833, 4.6667,
  5.25,  5.8333, 6.4167,  7.0,  7.5833, 8.1667, 8.75, 9.333, 9.9167,
  10.5,  11.0533, 11.6667, 15.2,  25.8,  36.5,
   RB(1) = 0.0,  0.1655, 0.3333, 0.4694, 0.6079, 0.7376, 0.8585, 0.9708,
  1.0745, 1.1693,  1.2563, 1.3346, 1.4045, 1.4661,  1.5194,  1.5645,
  1.6013, 1.6299, 1.6503, 1.6626, 1.6667, 1.6667, 1.6667, 1.6667,
   XWI(1,1) = -1.665, -1.665,  -11.995, -11.995,
 XWI(5,1) = -1.665, -1.665, -1.665, -3.731, -3.731, -3.731, -3.731,
 -3.731, -5.797, -5.797, -5.797, -5.797, -5.797, -7.863, -7.863, -7.863,
 -7.863, -7.863, -9.929, -9.929, -9.929, -9.929, -9.929, -11.995,
  -11.995, -11.995,
  YWI(1,1) = 0.0,  8.89,  14.05,  12.05,
  YWI(5,1) = 2.2225, 4.4450, 6.6675, 2.410, 4.288, 6.1661, 8.0441, 9.922,
 4.820, 6.3535, 7.887,  9.4204, 10.954, 7.230, 8.419, 9.608, 10.797,
 11.986, 9.640, 10.4844, 11.329, 12.1735, 13.018, 12.55, 13.05, 13.55,
 ZWI(5,1) = 0.1594, 0.1731, 0.1044,0.00, 0.1347, 0.1463, 0.0332, 0.0007,
 0.00, 0.11, 0.1195, 0.072, 0.0006, 0.00, 0.0852,  0.0926, 0.0558,
 0.0004,  0.00, 0.0605, 0.0658, 0.0397, 0.0003,  0.0358, 0.0389, 0.0235,
 $END
CORD1C         1               0.0       0.0      0.0     1.0     0.0      0.0
               1.      0.      1.0
CORD2R         2           15.59    0.0       0.0     15.59    0.0      1.0
        15.59      -1.0        1.0
       15           5
GRID          45         1    1.351 210.000    7.796        1        4
GRID          46         1    1.351 240.000    7.796        1        4
GRID          58         1    1.485 240.000    9.728        1        4
GRID          43         1    1.351 150.000    7.796        1        4
GRID          55         1    1.485 150.000    9.728        1        4
GRID          56         1    1.485 180.000    9.728        1        4
GRID          40         1    1.351  60.000    7.796        1        4
GRID          41         1    1.351  90.000    7.796        1        4
GRID          53         1    1.485  90.000    9.728        1        4
GRID          48         1    1.351 300.000    7.796        1        4
GRID          60         1    1.485 300.000    9.728        1        4
GRID          61         1    1.485 330.000    9.728        1        4
GRID          50         1    1.485   0.000    9.728        1        4
GRID          51         1    1.485  30.000    9.728        1        4
GRID          63         1    1.530  30.000   11.660        1        4
CTRIA2        90        90       40      41       53    0.000
CTRIA2        95        95       56      55       43    0.000
CTRIA2       100       100       45      46       58    0.000
CTRIA2       105       105       61      60       48    0.000
CTRIA2       110       110       50      51       63    0.000
       24          10
GRID         251         2   1.6650  -.0000   0.0000        2
GRID         252         2   1.6650  2.2225    .1594        2
GRID         261         2   3.7310  2.4100   0.0000        2
GRID         262         2   3.7310  4.2880    .1347        2
GRID         271         2   5.7970  4.8200   0.0000        2
GRID         272         2   5.7970  6.3535    .1100        2
GRID         281         2   7.8630  7.2300    .0000        2
GRID         282         2   7.8630  8.4190    .0852        2
GRID         291         2   9.9290  9.6400    .0000        2
GRID         292         2   9.9290 10.4844    .0605        2
GRID         301         2  11.9950 12.0500    .0000        2
GRID         302         2  11.9950 12.5500    .0358        2
GRID         254         2   1.6650  6.6675    .1044        2
GRID         255         2   1.6650  8.8900    .0000        2
GRID         264         2   3.7310  8.0441    .0382        2
GRID         265         2   3.7310  9.9220    .0007        2
```

GRID	274	2	5.7969	9.4204	.0720	2	
GRID	275	2	5.7970	10.9540	.0006	2	
GRID	284	2	7.8630	10.7970	.0558	2	
GRID	285	2	7.8630	11.9860	.0004	2	
GRID	294	2	9.9290	12.1735	.0397	2	
GRID	295	2	9.9290	13.0180	.0003	2	
GRID	304	2	11.9950	13.5500	.0235	2	
GRID	305	2	11.9950	14.0500	.0002	2	
CQUAD2	500	500	251	261	262	252	0.0000
CQUAD2	501	501	261	271	272	262	0.0000
CQUAD2	502	502	271	281	282	272	0.0000
CQUAD2	503	503	281	291	292	282	0.0000
CQUAD2	504	504	291	301	302	292	0.0000
CQUAD2	515	515	254	264	265	255	0.0000
CQUAD2	516	516	264	274	275	265	0.0000
CQUAD2	517	517	274	284	285	275	0.0000
CQUAD2	518	518	284	294	295	285	0.0000
CQUAD2	519	519	294	304	305	295	0.0000
	24	10					
GRID	251	2	1.6650	-.0000	0.0000	2	
GRID	312	2	1.6650	2.2225	-.1594	2	
GRID	321	2	3.7310	2.4100	0.0000	2	
GRID	322	2	3.7310	4.2800	-.1347	2	
GRID	331	2	5.7970	4.8200	0.0000	2	
GRID	332	2	5.7970	6.3535	-.1100	2	
GRID	341	2	7.8630	7.2300	-.0000	2	
GRID	342	2	7.8630	8.4190	-.0852	2	
GRID	351	2	9.9290	9.6400	-.0000	2	
GRID	352	2	9.9290	10.4844	-.0605	2	
GRID	361	2	11.9950	12.0500	-.0000	2	
GRID	362	2	11.9950	12.5500	-.0358	2	
GRID	314	2	1.6650	6.6675	-.1044	2	
GRID	255	2	1.6650	8.2900	.0000	2	
GRID	324	2	3.7310	8.0441	-.0882	2	
GRID	325	2	3.7310	9.9220	-.0007	2	
GRID	334	2	5.7969	9.4204	-.0720	2	
GRID	335	2	5.7970	10.9540	-.0006	2	
GRID	344	2	7.8630	10.7970	-.0558	2	
GRID	345	2	7.8630	11.9860	-.0004	2	
GRID	354	2	9.9290	12.1735	-.0397	2	
GRID	355	2	9.9290	13.0180	-.0003	2	
GRID	364	2	11.9950	13.5500	-.0235	2	
GRID	365	2	11.9950	14.0500	-.0002	2	
CQUAD2	525	525	251	321	322	312	0.0000
CQUAD2	526	526	321	331	332	322	0.0000
CQUAD2	527	527	331	341	342	332	0.0000
CQUAD2	528	528	341	351	352	342	0.0000
CQUAD2	529	529	351	361	362	352	0.0000
CQUAD2	540	540	314	324	325	255	0.0000
CQUAD2	541	541	324	334	335	325	0.0000
CQUAD2	542	542	334	344	345	335	0.0000
CQUAD2	543	543	344	354	355	345	0.0000
CQUAD2	544	544	354	364	365	355	0.0000

The input cards as shown, in sequence, are:

(I) Title Card

Notes: For input of any descriptive title

Format: (20A4)

(II) Namelist UNIFID input

Notes: For input of aerodynamic control parameters appropriate to the case.

Format: In accordance with the input variables whether real or integer.

Test Case:

ICASE = 3, complete missile, body and wing

NTRAN(1) = 1, (2) = 1, no intermediate coordinate system for wing and body.

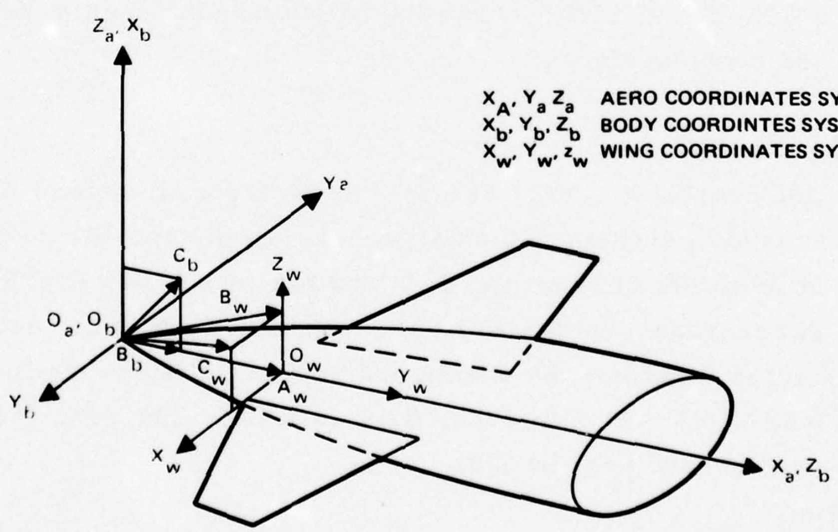
NBPFL = 24, number of points of input for body profile.

NWPI(1) = 30, number of points of input for wing surface, both upper and lower, given at same locations.

NWING = 1, one wing section.

ARB = 5.0, angle of attack of missile.

XMACH = 2.01, flow Mach number.



The above sketch shows the geometry of the complete missile with separate coordinate systems for body and wings. For body profile input, the radius RB and axial distance XB are given based on the cylindrical coordinate system oriented as shown. For wing plane form and surface contour input, XWI, YWI and ZWI representing the wing surface Z = f(x, y) are specified with ZWI as the height of wing surface. They can be given in either rectangular, cylindrical or spherical coordinate systems as desired. For the test case, we have for the body profile

XB(1) = 0.0, 0.5233, 1.1667, ...

RB(1) = 0.0, 0.1655, 0.3333, ...

For the wing surface, the wing plans are given as

XWI(1, 1) = -1.665, -1.665, -11.995, -11.995,

YWI(1, 1) = 0.0, 8.89, 14.05, 12.05,

For the first four points. The surface coordinates are

XWI(5, 1) = -1.665, -1.665, -1.665, -3.731, ...

YWI(5, 1) = 2.2225, 4.4450, 6.6675, 2.410, ...

ZWI(5, 1) = 0.1594, 0.1731, 0.1044, 0.0, ...

ZWI(5, 2) = -0.1594, -0.1731, -0.1044, 0.0, ...

where

ZWI(i, 1) and ZWI(i, 2) are the height of wing contour for upper and lower surfaces.

(III) Coordinate Card

Notes: The coordinate cards are needed for input of vectors A, B and C used in defining the relation between the coordinate systems of the finite element (X, Y, Z) and the aero codes (X_a, Y_a, Z_a). In accordance with NASTRAN, vector A defines the origin, vector B defines the Z axis and vector C defines the X-Z plane.

Format: NASTRAN bulk data deck, first card (A8, 2I8, 6F8.2) and second card (A8, 3F8.2).

Test Case:

Two sets of coordinate cards are required, one for the body and one for the wing. The body finite element coordinate system has its origin coincident with the aero code and its Z axis coincident with the X axis of the aero code as shown in the sketch. Hence,

the vector A has component (0.0, 0.0, 0.0), vector B has component (1.0, 0.0, 0.0) and vector C has component (1.0, 0.0, 1.0).

The finite element coordinate system for the wing as shown in the sketch requires the input of vector A with component (15.59, 0.0, 0.0), vector B (15.59, 0.0, 1.0) and vector C (15.59, -1.0, 1.0).

(IV) Finite Element Cards

Notes: The finite element and grid point cards as generated by BING and PING are used for defining element coordinates. They must be divided into two separate groups with the grid point card preceding the element card. A card specifies that the numbers of grid card and element card in a form 2I10 must be placed before them.

Format: For grid and element number card (2I10), grid and finite element cards must be the same as generated by BING or PING.

Test Case:

As shown in the input listing, first set of finite element grid point and element cards are that of the body and the subsequent two sets are for the upper and lower wing surfaces.

4.4 OUTPUT PRINTOUT OF TEST CASES

4.4.1 <u>Body Only Case</u>

FINAL TEST, BODY ALONE, NSWC CARD INPUT

```
$UNIFID
ICASE  = 2,
NTRANS = 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,   0, 0, 0,
IPUNCH = 0,
IRW    = 0,
POLAR  = 0.0,
ISOLID = 0,
SID    = 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
         0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
NBPFL  = 24,
XB     = 0.0, 0.5833E+00, 0.11667E+01, 0.175E+01, 0.2333E+01,             0.2916E+01, 0.35E+01, 0.40833E+01,
         0.46667E+01, 0.525E+01, 0.58333E+01, 0.64167E+01, 0.7E+01,       0.75833E+01, 0.81667E+01,
         0.875E+01, 0.9333E+01, 0.99167E+01, 0.105E+02, 0.110533E+02,     0.116667E+02, 0.152E+02,
         0.258E+02, 0.365E+02, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0,
RB     = 0.0, 0.1655E+00, 0.3333E+00, 0.4694E+00, 0.6079E+00,             0.7376E+00, 0.8585E+00, 0.9708E+00,
         0.10745E+01, 0.11696E+01, 0.12563E+01, 0.13346E+01, 0.14045E+01, 0.14661E+01, 0.15194E+01,
         0.15645E+01, 0.16013E+01, 0.16299E+01, 0.16503E+01, 0.16626E+01, 0.16667E+01, 0.16667E+01,
         0.16667E+01, 0.16667E+01, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0,
ZD     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0,
RP     = 0.0,
NWPI   = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,   0, 0, 0,
NWING  = 0,
IFORM  = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
PIVOT  = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
THICK  = 0.1E+01,
DIHED  = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XWI    = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,           0.0. 0.0. 0.0. 0.0. 0.0.
```

```
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

YWI = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

ZWI = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
```

```
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XNACEL  = 0.0,
NACELP  = 0,
XN      = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
RN      = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
ZN      = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
```

```
XMACH    =  0.201E+01,
PINF     =  0.147E+02,
DADEG    =  0.0,
ARB      =  0.5E+01,
ARW      =  0.0,
ARN      =  0.0,  0.0,
$END
EXIT INTAPE

DESCRIPTION OF CASE REQUESTED

SYMMETRICAL CONFIGURATION - PANELS LOCATED ON BOTH SIDES OF X-Z PLANE(SYM = 1.)
CASE = 2.          CALCULATE CL, GIVEN SHAPE
CPCALC = 0.        LINEAR CP
POLAR = 1.         POLARS REQUESTED
THICK = 0.         WING THICKNESS PRESSURES NOT TO BE ADDED
VOUT = 0.          VELOCITY COMPONENTS NOT TO BE PRINTED
MACH NUMBER =    2.0100
POINT ABOUT WHICH THE MOMENTS ARE TO BE COMPUTED
 X-COORDINATE =    0.0000
 Z-COORDINATE =    0.0000
REFERENCE CHORD LENGTH =    1.0000
WING REFERENCE AREA =    1.0000
WING SEMI-SPAN =     1.0000
BODY REFERENCE AREA  =      8.7469
HEIGHT OF WING PLANE ABOVE BODY AXIS =    0.0000
INCLINATION OF BODY AXIS WITH RESPECT TO DEFINING AXIS =    0.0000  DEG.
ANGLE OF ATTACK WITH RESPECT TO BODY AXIS =    5.0000  DEG.
```

PRESSURES, FORCES AND MOMENTS ON BODY

CD = .06332 CL = .13074 CM = -.73702

BODY PRESSURE COEFFICIENTS(CP)

THETA(DEG.) X	0.0000	30.0000	60.0000	90.0000	120.0000	150.0000	180.0000
0.0000	.12638	.13607	.16616	.21578	.27524	.32501	.34454
.7300	.11196	.12129	.15040	.19876	.25703	.30596	.32520
1.4600	.08963	.09797	.12453	.16979	.22539	.27265	.29134
2.1900	.08156	.08911	.11368	.15656	.21022	.25633	.27465
2.9200	.06710	.07400	.09689	.13771	.18955	.23448	.25241
3.6500	.05376	.05992	.08089	.11935	.16914	.21278	.23027
4.3800	.04049	.04587	.06482	.10079	.14842	.19067	.20769
5.1100	.02795	.03250	.04931	.08266	.12800	.16878	.18531
5.8400	.01576	.01944	.03401	.06459	.10749	.14671	.16272
6.5700	.00414	.00691	.01912	.04677	.08710	.12467	.14011
7.3000	-.01976	-.01824	-.00931	.01422	.05081	.08588	.10047
9.2467	-.04166	-.04303	-.04159	-.02744	.00080	.03041	.04313
11.1933	-.04718	-.05153	-.05788	-.05347	-.03395	-.01008	.00068
13.1400	-.02317	-.02981	-.04214	-.04519	-.03229	-.01275	-.00346
15.0867	-.00714	-.01505	-.03072	-.03808	-.02921	-.01244	-.00413
17.0333	.00291	-.00554	-.02270	-.03210	-.02531	-.01006	-.00231
18.9800	.00802	-.00052	-.01798	-.02795	-.02190	-.00730	.00019
20.9267	.01049	.00208	-.01512	-.02491	-.01887	-.00442	.00299
22.8733	.01146	.00324	-.01348	-.02279	-.01645	-.00189	.00553
24.8200	.01175	.00372	-.01254	-.02133	-.01462	.00011	.00759
26.7667	.01184	.00395	-.01195	-.02032	-.01330	.00161	.00913
28.7133	.01195	.00417	-.01148	-.01956	-.01230	.00274	.01030
30.6600	.01213	.00441	-.01108	-.01899	-.01159	.00354	.01112
32.6067	.01236	.00467	-.01074	-.01855	-.01107	.00410	.01170
34.5533	.01263	.00495	-.01043	-.01821	-.01071	.00447	.01207
36.5000	.01329	.00562	-.00973	-.01747	-.00993	.00526	.01287

AERODYNAMIC PRESSURE ON THE BODY - - CENTROID OF FINITE ELEMENT AS GENERATED BY NSWC CODE IN THE SAME LOCAL COORDINATE SYSTEM

ELEMENT	X	THETA	PRESSURE	ELEMENT	X	THETA	PRESSURE
90	.844E+01	.280E+03	.480E+00	95	.908E+01	.200E+03	.322E+01
100	.844E+01	.130E+03	.305E+01	105	.908E+01	.498E+02	.157E+00
110	.104E+02	.340E+03	-.541E+00				

.........END OF COMPUTATIONS ....

4.4.2 Wing Only Case

FINAL TEST CASE, WING ONLY, NASA MEMO 10 - 15 - 58L

```
$UNIFID
ICASE  = 1,
NTRANS = 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0,
IPUNCH = 0,
IRW    = 0,
POLAR  = 0.0,
ISOLID = 0,
IFORM  = 1, 0, 0, 0, 0, 0, 0, 0, 0, 0,
NBPFL  = 0,
XB     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0,
RB     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0,
ZD     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0,
RP     = 0.0,
NWPI   = 30, 30, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
NWING  = 1,
PIVOT  = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
THICK  = 0.1E+01,
DIHED  = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XWI    =-0.1665E+01, -0.1665E+01, -0.11995E+02, -0.11995E+02, -0.1665E+01,   -0.1665E+01, -0.1665E+01, -0.3731E+01,
        -0.3731E+01, -0.3731E+01, -0.3731E+01, -0.3731E+01, -0.5797E+01,   -0.5797E+01, -0.5797E+01, -0.5797E+01,
        -0.5797E+01, -0.7863E+01, -0.7863E+01, -0.7863E+01, -0.7863E+01,   -0.7863E+01, -0.9929E+01, -0.9929E+01,
        -0.9929E+01, -0.9929E+01, -0.9929E+01, -0.11995E+02, -0.11995E+02,   -0.11995E+02, -0.1665E+01,
        -0.1665E+01, -0.11995E+02, -0.11995E+02, -0.1765E+01, -0.1765E+01,   -0.1765E+01, -0.3831E+01, -0.3831E+01,
        -0.3831E+01, -0.3831E+01, -0.3831E+01, -0.5897E+01, -0.5897E+01,   -0.5897E+01, -0.5897E+01, -0.5897E+01,
        -0.7963E+01, -0.7963E+01, -0.7963E+01, -0.7963E+01, -0.7963E+01,   -0.9999E+01, -0.9999E+01, -0.9999E+01,
        -0.9999E+01, -0.9999E+01, -0.11895E+02, -0.11895E+02, -0.11895E+02,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
```


MCDONNELL DOUGLAS

```
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
YWI = 0.0, 0.889E+01, 0.1405E+02, 0.1205E+02, 0.22225E+01,         0.4445E+01, 0.66675E+01,
      0.241E+01, 0.4288E+01, 0.61661E+01, 0.80441E+01, 0.9922E+01,  0.482E+01, 0.63535E+01, 0.7887E+01,
      0.94204E+01, 0.10954E+02, 0.723E+01, 0.8419E+01, 0.9608E+01,  0.10797E+02, 0.11986E+02,
      0.964E+01, 0.104844E+02, 0.11329E+02, 0.121735E+02, 0.13018E+02, 0.1255E+02, 0.1305E+02, 0.1355E+02,
      0.0, 0.889E+01, 0.1405E+02, 0.1205E+02, 0.21225E+01,          0.4345E+01, 0.6567E+01, 0.251E+01,
      0.4388E+01, 0.6266E+01, 0.8144E+01, 0.994E+01, 0.492E+01,     0.645E+01, 0.798E+01, 0.945E+01,
      0.11E+02, 0.733E+01, 0.8519E+01, 0.9708E+01, 0.10897E+02,     0.11996E+02, 0.976E+01, 0.1058E+02,
      0.1142E+02, 0.1227E+02, 0.13118E+02, 0.1255E+02, 0.1315E+02,  0.1355E+02, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
      0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
```

```
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

ZWI    = 0.0, 0.0, 0.0, 0.0, 0.1594E+00, 0.1731E+00, 0.1044E+00, 0.0,  0.1347E+00, 0.1463E+00, 0.882E-01,
         0.7E-03, 0.0, 0.11E+00, 0.1195E+00, 0.72E-01, 0.6E-03, 0.0,   0.852E-01, 0.926E-01, 0.558E-01,
         0.4E-03, 0.0, 0.605E-01, 0.658E-01, 0.397E-01, 0.3E-03,       0.358E-01, 0.389E-01, 0.235E-01,
         0.0, 0.0, 0.0, 0.0, -0.1594E+00, -0.1731E+00, -0.1044E+00, 0.0, -0.1347E+00, -0.1463E+00, -0.882E-01,
        -0.7E-03, 0.0, -0.11E+00, -0.1195E+00, -0.72E-01, -0.6E-03, -0.0, -0.852E-01, -0.926E-01, -0.558E-01,
        -0.4E-03, 0.0, -0.605E-01, -0.658E-01, -0.397E-01, -0.3E-03,   -0.358E-01, -0.389E-01, -0.235E-01,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0,

XNACEL = 0.0,

NACELP = 0,

XN     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

RN     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
```

```
ZN     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

XMACH  = 0.201E+01,

PINF   = 0.147E+02,

DADEG  = 0.0,

ARB    = 0.0,

ARW    = 0.5E+01,

ARN    = 0.0, 0.0,

SID    = -0.1E+01, -0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
          0.1E+01,  0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,

$END
```

WING PANEL CORNER POINT COORDINATES
1 AND 2 INDICATE WING PANEL LEADING-EDGE POINTS, 3 AND 4 INDICATE TRAILING-EDGE POINTS

WING NO. 1 YPIVOT = 1.665 ZPIVOT = 0.000 WING DIHEDRAL = 0.000 RADIANS

PANEL NO	PARTS	X_1	Y_1	Z_1	X_2	Y_2	Z_2	X_3	Y_3	Z_3	X_4	Y_4	Z_4
1	1	15.59000	1.66500	0.00000	18.00000	3.73100	0.00000	16.47900	1.66500	0.00000	18.75120	3.73100	0.00000
2	1	16.47900	1.66500	0.00000	18.75120	3.73100	0.00000	17.36800	1.66500	0.00000	19.50240	3.73100	0.00000
3	1	17.36800	1.66500	0.00000	19.50240	3.73100	0.00000	18.25700	1.66500	0.00000	20.25360	3.73100	0.00000
4	1	18.25700	1.66500	0.00000	20.25360	3.73100	0.00000	19.14600	1.66500	0.00000	21.00480	3.73100	0.00000
5	1	19.14600	1.66500	0.00000	21.00480	3.73100	0.00000	20.03500	1.66500	0.00000	21.75600	3.73100	0.00000
6	1	20.03500	1.66500	0.00000	21.75600	3.73100	0.00000	20.92400	1.66500	0.00000	22.50720	3.73100	0.00000
7	1	20.92400	1.66500	0.00000	22.50720	3.73100	0.00000	21.81300	1.66500	0.00000	23.25840	3.73100	0.00000
8	1	21.81300	1.66500	0.00000	23.25840	3.73100	0.00000	22.70200	1.66500	0.00000	24.00960	3.73100	0.00000
9	1	22.70200	1.66500	0.00000	24.00960	3.73100	0.00000	23.59100	1.66500	0.00000	24.76080	3.73100	0.00000
10	1	23.59100	1.66500	0.00000	24.76080	3.73100	0.00000	24.48000	1.66500	0.00000	25.51200	3.73100	0.00000
11	1	18.00000	3.73100	0.00000	20.41000	5.79700	0.00000	18.75120	3.73100	0.00000	21.02340	5.79700	0.00000
12	1	18.75120	3.73100	0.00000	21.02340	5.79700	0.00000	19.50240	3.73100	0.00000	21.63680	5.79700	0.00000
13	1	19.50240	3.73100	0.00000	21.63680	5.79700	0.00000	20.25360	3.73100	0.00000	22.25020	5.79700	0.00000
14	1	20.25360	3.73100	0.00000	22.25020	5.79700	0.00000	21.00480	3.73100	0.00000	22.86360	5.79700	0.00000
15	1	21.00480	3.73100	0.00000	22.86360	5.79700	0.00000	21.75600	3.73100	0.00000	23.47700	5.79700	0.00000
16	1	21.75600	3.73100	0.00000	23.47700	5.79700	0.00000	22.50720	3.73100	0.00000	24.09040	5.79700	0.00000
17	1	22.50720	3.73100	0.00000	24.09040	5.79700	0.00000	23.25840	3.73100	0.00000	24.70380	5.79700	0.00000
18	1	23.25840	3.73100	0.00000	24.70380	5.79700	0.00000	24.00960	3.73100	0.00000	25.31720	5.79700	0.00000
19	1	24.00960	3.73100	0.00000	25.31720	5.79700	0.00000	24.76080	3.73100	0.00000	25.93060	5.79700	0.00000
20	1	24.76080	3.73100	0.00000	25.93060	5.79700	0.00000	25.51200	3.73100	0.00000	26.54400	5.79700	0.00000
21	1	20.41000	5.79700	0.00000	22.82000	7.86300	0.00000	21.02340	5.79700	0.00000	23.29560	7.86300	0.00000
22	1	21.02340	5.79700	0.00000	23.29560	7.86300	0.00000	21.63680	5.79700	0.00000	23.77120	7.86300	0.00000
23	1	21.63680	5.79700	0.00000	23.77120	7.86300	0.00000	22.25020	5.79700	0.00000	24.24680	7.86300	0.00000
24	1	22.25020	5.79700	0.00000	24.24680	7.86300	0.00000	22.86360	5.79700	0.00000	24.72240	7.86300	0.00000
25	1	22.86360	5.79700	0.00000	24.72240	7.86300	0.00000	23.47700	5.79700	0.00000	25.19800	7.86300	0.00000
26	1	23.47700	5.79700	0.00000	25.19800	7.86300	0.00000	24.09040	5.79700	0.00000	25.67360	7.86300	0.00000
27	1	24.09040	5.79700	0.00000	25.67360	7.86300	0.00000	24.70380	5.79700	0.00000	26.14920	7.86300	0.00000
28	1	24.70380	5.79700	0.00000	26.14920	7.86300	0.00000	25.31720	5.79700	0.00000	26.62480	7.86300	0.00000
29	1	25.31720	5.79700	0.00000	26.62480	7.86300	0.00000	25.93060	5.79700	0.00000	27.10040	7.86300	0.00000
30	1	25.93060	5.79700	0.00000	27.10040	7.86300	0.00000	26.54400	5.79700	0.00000	27.57600	7.86300	0.00000
31	1	22.82000	7.86300	0.00000	25.23000	9.92900	0.00000	23.29560	7.86300	0.00000	25.56780	9.92900	0.00000
32	1	23.29560	7.86300	0.00000	25.56780	9.92900	0.00000	23.77120	7.86300	0.00000	25.90560	9.92900	0.00000
33	1	23.77120	7.86300	0.00000	25.90560	9.92900	0.00000	24.24680	7.86300	0.00000	26.24340	9.92900	0.00000
34	1	24.24680	7.86300	0.00000	26.24340	9.92900	0.00000	24.72240	7.86300	0.00000	26.58120	9.92900	0.00000
35	1	24.72240	7.86300	0.00000	26.58120	9.92900	0.00000	25.19800	7.86300	0.00000	26.91900	9.92900	0.00000
36	1	25.19800	7.86300	0.00000	26.91900	9.92900	0.00000	25.67360	7.86300	0.00000	27.25680	9.92900	0.00000
37	1	25.67360	7.86300	0.00000	27.25680	9.92900	0.00000	26.14920	7.86300	0.00000	27.59460	9.92900	0.00000
38	1	26.14920	7.86300	0.00000	27.59460	9.92900	0.00000	26.62480	7.86300	0.00000	27.93240	9.92900	0.00000
39	1	26.62480	7.86300	0.00000	27.93240	9.92900	0.00000	27.10040	7.86300	0.00000	28.27020	9.92900	0.00000
40	1	27.10040	7.86300	0.00000	28.27020	9.92900	0.00000	27.57600	7.86300	0.00000	28.60800	9.92900	0.00000
41	1	25.23000	9.92900	0.00000	27.64000	11.99500	0.00000	25.56780	9.92900	0.00000	27.84000	11.99500	0.00000
42	1	25.56780	9.92900	0.00000	27.84000	11.99500	0.00000	25.90560	9.92900	0.00000	28.04000	11.99500	0.00000
43	1	25.90560	9.92900	0.00000	28.04000	11.99500	0.00000	26.24340	9.92900	0.00000	28.24000	11.99500	0.00000
44	1	26.24340	9.92900	0.00000	28.24000	11.99500	0.00000	26.58120	9.92900	0.00000	28.44000	11.99500	0.00000
45	1	26.58120	9.92900	0.00000	28.44000	11.99500	0.00000	26.91900	9.92900	0.00000	28.64000	11.99500	0.00000
46	1	26.91900	9.92900	0.00000	28.64000	11.99500	0.00000	27.25680	9.92900	0.00000	28.84000	11.99500	0.00000
47	1	27.25680	9.92900	0.00000	28.84000	11.99500	0.00000	27.59460	9.92900	0.00000	29.04000	11.99500	0.00000
48	1	27.59460	9.92900	0.00000	29.04000	11.99500	0.00000	27.93240	9.92900	0.00000	29.24000	11.99500	0.00000
49	1	27.93240	9.92900	0.00000	29.24000	11.99500	0.00000	28.27020	9.92900	0.00000	29.44000	11.99500	0.00000
50	1	28.27020	9.92900	0.00000	29.44000	11.99500	0.00000	28.60800	9.92900	0.00000	29.64000	11.99500	0.00000

WING PANEL CENTROID AND CONTROL POINT COORDINATES

PANEL	X_C	Y_C	Z_C	X_{CP}	Y_{CP}	Z_{CP}	AREA	THETA-DIHED	ALPHA-CAMBER	CHORD
1	17.17227	2.66907	0.00000	17.54218	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
2	17.99430	2.66907	0.00000	18.36421	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
3	18.81633	2.66907	0.00000	19.18624	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
4	19.63836	2.66907	0.00000	20.00827	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
5	20.46039	2.66907	0.00000	20.83030	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
6	21.28242	2.66907	0.00000	21.65233	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
7	22.10445	2.66907	0.00000	22.47436	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
8	22.92648	2.66907	0.00000	23.29639	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
9	23.74851	2.66907	0.00000	24.11842	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
10	24.57053	2.66907	0.00000	24.94045	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
11	19.50675	4.72923	0.00000	19.81483	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
12	20.19137	4.72923	0.00000	20.49945	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
13	20.87599	4.72923	0.00000	21.18407	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
14	21.56061	4.72923	0.00000	21.86868	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
15	22.24523	4.72923	0.00000	22.55330	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
16	22.92984	4.72923	0.00000	23.23792	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
17	23.61446	4.72923	0.00000	23.92254	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
18	24.29908	4.72923	0.00000	24.60716	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
19	24.98370	4.72923	0.00000	25.29178	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
20	25.66832	4.72923	0.00000	25.97640	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
21	21.83788	6.78643	0.00000	22.08421	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
22	22.38528	6.78643	0.00000	22.63162	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
23	22.93269	6.78643	0.00000	23.17902	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
24	23.48010	6.78643	0.00000	23.72643	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
25	24.02750	6.78643	0.00000	24.27383	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
26	24.57491	6.78643	0.00000	24.82124	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
27	25.12231	6.78643	0.00000	25.36865	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
28	25.66972	6.78643	0.00000	25.91605	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
29	26.21713	6.78643	0.00000	26.46346	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
30	26.76453	6.78643	0.00000	27.01087	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
31	24.16225	8.83767	0.00000	24.34701	8.83767	0.00000	.84024	0.00000	0.00000	.41059
32	24.57284	8.83767	0.00000	24.75760	8.83767	0.00000	.84024	0.00000	0.00000	.41059
33	24.98343	8.83767	0.00000	25.16820	8.83767	0.00000	.84024	0.00000	0.00000	.41059
34	25.39402	8.83767	0.00000	25.57879	8.83767	0.00000	.84024	0.00000	0.00000	.41059
35	25.80461	8.83767	0.00000	25.98938	8.83767	0.00000	.84024	0.00000	0.00000	.41059
36	26.21520	8.83767	0.00000	26.39997	8.83767	0.00000	.84024	0.00000	0.00000	.41059
37	26.62579	8.83767	0.00000	26.81056	8.83767	0.00000	.84024	0.00000	0.00000	.41059
38	27.03638	8.83767	0.00000	27.22115	8.83767	0.00000	.84024	0.00000	0.00000	.41059
39	27.44697	8.83767	0.00000	27.63174	8.83767	0.00000	.84024	0.00000	0.00000	.41059
40	27.85757	8.83767	0.00000	28.04233	8.83767	0.00000	.84024	0.00000	0.00000	.41059
41	26.46947	10.87377	0.00000	26.59313	10.87377	0.00000	.55555	0.00000	0.00000	.27478
42	26.74426	10.87377	0.00000	26.86791	10.87377	0.00000	.55555	0.00000	0.00000	.27478
43	27.01904	10.87377	0.00000	27.14270	10.87377	0.00000	.55555	0.00000	0.00000	.27478
44	27.29383	10.87377	0.00000	27.41748	10.87377	0.00000	.55555	0.00000	0.00000	.27478
45	27.56861	10.87377	0.00000	27.69227	10.87377	0.00000	.55555	0.00000	0.00000	.27478
46	27.84340	10.87377	0.00000	27.96705	10.87377	0.00000	.55555	0.00000	0.00000	.27478
47	28.11818	10.87377	0.00000	28.24184	10.87377	0.00000	.55555	0.00000	0.00000	.27478
48	28.39297	10.87377	0.00000	28.51662	10.87377	0.00000	.55555	0.00000	0.00000	.27478
49	28.66775	10.87377	0.00000	28.79140	10.87377	0.00000	.55555	0.00000	0.00000	.27478
50	28.94254	10.87377	0.00000	29.06619	10.87377	0.00000	.55555	0.00000	0.00000	.27478

EXIT INTAPE

EXIT EVAL

EXIT EVAL

```
DESCRIPTION OF CASE REQUESTED

SYMMETRICAL CONFIGURATION - PANELS LOCATED ON BOTH SIDES OF X-Z PLANE(SYM = 1.)

CASE = 2.          CALCULATE CL, GIVEN SHAPE

CPCALC = 0.        LINEAR CP

POLAR = 1.         POLARS REQUESTED

THICK = 1.         WING THICKNESS PRESSURES TO BE ADDED

VOUT = 0.          VELOCITY COMPONENTS NOT TO BE PRINTED

MACH NUMBER =    2.0100

POINT ABOUT WHICH THE MOMENTS ARE TO BE COMPUTED
 X-COORDINATE =    0.0000
 Z-COORDINATE =    0.0000

REFERENCE CHORD LENGTH =    1.0000

WING REFERENCE AREA =   56.2468

WING SEMI-SPAN =     1.0000

INCLINATION OF BODY AXIS WITH RESPECT TO DEFINING AXIS =    0.0000  DEG.

ANGLE OF ATTACK WITH RESPECT TO BODY AXIS =    0.0000  DEG.
```

PRESSURES, FORCES AND MOMENTS ON WING

CD = .01754 CL = .19380 CM = -4.59133

SECTION CD DISTRIBUTION

SPANWISE STATION	1	2	3	4	5
	.01050	.01691	.02166	.02333	.02350

SECTION CL DISTRIBUTION

SPANWISE STATION	1	2	3	4	5
	.11439	.17955	.23880	.26761	.26936

UPPER SURFACE WING PANEL PRESSURE COEFFICIENTS(CP)

SPANWISE STATION CHORDWISE STATION	1	2	3	4	5
1	-.09520	-.12815	-.17064	-.14518	-.12249
2	-.07904	-.13419	-.11030	-.14825	-.12256
3	-.05805	-.13073	-.07121	-.15368	-.12905
4	-.04222	-.12393	-.10727	-.13756	-.13233
5	-.03439	-.10557	-.12046	-.11620	-.13393
6	-.03220	-.05241	-.12325	-.10530	-.13493
7	-.03326	.00698	-.11868	-.10919	-.13808
8	-.03514	-.00639	-.10627	-.12152	-.14236
9	-.03455	-.01576	-.09297	-.12344	-.14301
10	-.01433	-.02802	-.08548	-.12382	-.13603

LOWER SURFACE WING PANEL PRESSURE COEFFICIENTS(CP)

SPANWISE STATION CHORDWISE STATION	1	2	3	4	5
1	.16372	.14111	.09872	.12419	.14687
2	.11627	.13300	.15905	.12112	.14680
3	.08835	.12128	.19793	.11568	.14031
4	.06937	.09609	.16030	.13179	.13703
5	.05316	.08154	.14241	.15312	.13543
6	.04206	.10496	.12825	.16388	.13443
7	.03523	.13944	.11376	.15948	.13128
8	.03126	.10728	.10433	.14578	.12700
9	.03263	.08573	.09522	.14115	.12634
10	.05348	.06691	.08151	.13575	.13331

WING PANEL PRESSURE DIFFERENCE(CL)

SPANWISE STATION CHORDWISE STATION	1	2	3	4	5
1	.25893	.26926	.26936	.26936	.26936
2	.19531	.26720	.26935	.26936	.26936
3	.14640	.25201	.26914	.26936	.26936
4	.11159	.22002	.26758	.26936	.26936
5	.08755	.18711	.26287	.26933	.26936
6	.07426	.15737	.25149	.26918	.26936
7	.06849	.13245	.23244	.26867	.26936
8	.06640	.11367	.21060	.26730	.26936
9	.06718	.10149	.18819	.26459	.26935
10	.06781	.09493	.16699	.25957	.26933

UPPER SURFACE WING PANEL SLOPE(DZ/DX)

SPANWISE STATION / CHORDWISE STATION	1	2	3	4	5
1	-.06507	-.08307	-.11057	-.09407	-.07937
2	-.06492	-.09497	-.08042	-.09327	-.07942
3	-.06487	-.10477	-.05247	-.09252	-.07957
4	-.06487	-.11022	-.05787	-.09182	-.07982
5	-.06497	-.11137	-.06387	-.09112	-.08012
6	-.06517	-.08062	-.06882	-.09042	-.08047
7	-.06537	-.05182	-.07357	-.08977	-.08092
8	-.06567	-.05552	-.08097	-.08837	-.08147
9	-.06607	-.06117	-.09112	-.08727	-.08207
10	-.05832	-.06887	-.10052	-.08727	-.08267

LOWER SURFACE WING PANEL SLOPE(DZ/DX)

SPANWISE STATION / CHORDWISE STATION	1	2	3	4	5
1	-.10947	-.09147	-.06397	-.08047	-.09517
2	-.10962	-.07957	-.09412	-.08127	-.09512
3	-.10967	-.06977	-.12207	-.08202	-.09497
4	-.10967	-.06432	-.11667	-.08272	-.09472
5	-.10957	-.06317	-.11067	-.08342	-.09442
6	-.10937	-.09392	-.10572	-.08412	-.09407
7	-.10917	-.12272	-.10097	-.08477	-.09362
8	-.10887	-.11902	-.09357	-.08617	-.09307
9	-.10847	-.11337	-.08342	-.08727	-.09247
10	-.11622	-.10567	-.07402	-.08727	-.09187

WING CAMBER SLOPE(DZ/DX)

SPANWISE STATION / CHORDWISE STATION	1	2	3	4	5
1	-.08727	-.08727	-.08727	-.08727	-.08727
2	-.08727	-.08727	-.08727	-.08727	-.08727
3	-.08727	-.08727	-.08727	-.08727	-.08727
4	-.08727	-.08727	-.08727	-.08727	-.08727
5	-.08727	-.08727	-.08727	-.08727	-.08727
6	-.08727	-.08727	-.08727	-.08727	-.08727
7	-.08727	-.08727	-.08727	-.08727	-.08727
8	-.08727	-.08727	-.08727	-.08727	-.08727
9	-.08727	-.08727	-.08727	-.08727	-.08727
10	-.08727	-.08727	-.08727	-.08727	-.08727

WING CHORD LENGTHS(C)

8.22030	6.84619	5.47406	4.10591	2.74785

PRESSURE AT CONTROL POINT - UPPER SURFACE

X	Y	PRESSURE
17.5422	2.6691	-.3958E+01
18.3642	2.6691	-.3286E+01
19.1862	2.6691	-.2413E+01
20.0083	2.6691	-.1755E+01
20.8303	2.6691	-.1430E+01
21.6523	2.6691	-.1339E+01
22.4744	2.6691	-.1383E+01
23.2964	2.6691	-.1461E+01

24.1184	2.6691	-.1436E+01
24.9404	2.6691	-.5957E+00
19.8148	4.7292	-.5327E+01
20.4994	4.7292	-.5579E+01
21.1841	4.7292	-.5435E+01
21.8687	4.7292	-.5152E+01
22.5533	4.7292	-.4389E+01
23.2379	4.7292	-.2179E+01
23.9225	4.7292	.2902E+00
24.6072	4.7292	-.2657E+00
25.2918	4.7292	-.6554E+00
25.9764	4.7292	-.1165E+01
22.0842	6.7864	-.7094E+01
22.6316	6.7864	-.4585E+01
23.1790	6.7864	-.2960E+01
23.7264	6.7864	-.4460E+01
24.2738	6.7864	-.5008E+01
24.8212	6.7864	-.5124E+01
25.3686	6.7864	-.4934E+01
25.9161	6.7864	-.4410E+01
26.4635	6.7864	-.3865E+01
27.0109	6.7864	-.3553E+01
24.3470	8.8377	-.6035E+01
24.7576	8.8377	-.6163E+01
25.1682	8.8377	-.6389E+01
25.5788	8.8377	-.5719E+01
25.9894	8.8377	-.4831E+01
26.4000	8.8377	-.4377E+01
26.8106	8.8377	-.4539E+01
27.2211	8.8377	-.5052E+01
27.6317	8.8377	-.5132E+01
28.0423	8.8377	-.5147E+01
26.5931	10.8738	-.5092E+01
26.8679	10.8738	-.5095E+01
27.1427	10.8738	-.5365E+01
27.4175	10.8738	-.5501E+01
27.6923	10.8738	-.5568E+01
27.9671	10.8738	-.5610E+01
28.2418	10.8738	-.5741E+01
28.5166	10.8738	-.5918E+01
28.7914	10.8738	-.5945E+01
29.0662	10.8738	-.5655E+01

PRESSURE AT CONTROL POINT - LOWER SURFACE

X	Y	PRESSURE
17.5422	2.6691	.6806E+01
18.3642	2.6691	.4834E+01
19.1862	2.6691	.3673E+01
20.0083	2.6691	.2884E+01
20.8303	2.6691	.2210E+01
21.6523	2.6691	.1749E+01
22.4744	2.6691	.1465E+01
23.2964	2.6691	.1300E+01
24.1184	2.6691	.1356E+01
24.9404	2.6691	.2223E+01
19.8148	4.7292	.5866E+01
20.4994	4.7292	.5529E+01
21.1841	4.7292	.5042E+01
21.8687	4.7292	.3995E+01
22.5533	4.7292	.3390E+01
23.2379	4.7292	.4364E+01
23.9225	4.7292	.5797E+01

24.6072	4.7292	.4460E+01
25.2918	4.7292	.3564E+01
25.9764	4.7292	.2782E+01
22.0842	6.7864	.4104E+01
22.6316	6.7864	.6612E+01
23.1790	6.7864	.8228E+01
23.7264	6.7864	.6664E+01
24.2738	6.7864	.5921E+01
24.8212	6.7864	.5332E+01
25.3686	6.7864	.4729E+01
25.9161	6.7864	.4337E+01
26.4635	6.7864	.3958E+01
27.0109	6.7864	.3389E+01
24.3470	8.8377	.5163E+01
24.7576	8.8377	.5035E+01
25.1682	8.8377	.4809E+01
25.5788	8.8377	.5479E+01
25.9894	8.8377	.6366E+01
26.4000	8.8377	.6813E+01
26.8106	8.8377	.6630E+01
27.2211	8.8377	.6061E+01
27.6317	8.8377	.5868E+01
28.0423	8.8377	.5643E+01
26.5931	10.8738	.6106E+01
26.8679	10.8738	.6103E+01
27.1427	10.8738	.5833E+01
27.4175	10.8738	.5697E+01
27.6923	10.8738	.5630E+01
27.9671	10.8738	.5589E+01
28.2418	10.8738	.5457E+01
28.5166	10.8738	.5280E+01
28.7914	10.8738	.5252E+01
29.0662	10.8738	.5542E+01

AERODYNAMIC PRESSURE ON THE WING - - CENTROID OF FINITE ELEMENT AS GENERATED BY NSWC CODE

TOTAL NUMBER OF WINGS 1

WING NO. 1

PRESSURE LOAD ON THE UPPER SURFACE OF WING

ELEMENT	X	Y	PRESSURE	ELEMENT	X	Y	PRESSURE
500	.178E+02	-.267E+01	.377E+01	501	.200E+02	-.473E+01	.539E+01
502	.222E+02	-.679E+01	.640E+01	503	.245E+02	-.884E+01	.606E+01
504	.267E+02	-.109E+02	.508E+01	515	.240E+02	-.267E+01	.149E+01
516	.252E+02	-.473E+01	.586E+00	517	.264E+02	-.679E+01	.396E+01
518	.275E+02	-.884E+01	.513E+01	519	.287E+02	-.109E+02	.596E+01

PRESSURE LOAD ON THE LOWER SURFACE OF WING

ELEMENT	X	Y	PRESSURE	ELEMENT	X	Y	PRESSURE
525	.178E+02	-.267E+01	-.625E+01	526	.200E+02	-.473E+01	-.577E+01
527	.222E+02	-.679E+01	-.476E+01	528	.245E+02	-.884E+01	-.513E+01
529	.267E+02	-.109E+02	-.612E+01	540	.240E+02	-.267E+01	-.129E+01
541	.252E+02	-.473E+01	-.372E+01	542	.264E+02	-.679E+01	-.405E+01
543	.275E+02	-.884E+01	-.590E+01	544	.287E+02	-.109E+02	-.523E+01

.......END OF COMPUTATIONS ....

4.4.3 Complete Missile Case

FINAL TEST CASE, COMPLETE MISSILE, NASA MEMO 10 - 15 - 58L

```
$UNIFID
ICASE  = 3,
NTRANS = 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0,
IPUNCH = 0,
IRW    = 0,
POLAR  = 0.0,
ISOLID = 0,
SID    = 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
         0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01, 0.1E+01,
NBPFL  = 24,
XB     = 0.0, 0.5833E+00, 0.11667E+01, 0.175E+01, 0.2333E+01,          0.2916E+01, 0.35E+01, 0.40833E+01,
         0.46667E+01, 0.525E+01, 0.58333E+01, 0.64167E+01, 0.7E+01,    0.75833E+01, 0.81667E+01,
         0.875E+01, 0.9333E+01, 0.99167E+01, 0.105E+02, 0.110533E+02,   0.116667E+02, 0.152E+02,
         0.258E+02, 0.365E+02, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,       0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0,
RB     = 0.0, 0.1655E+00, 0.3333E+00, 0.4694E+00, 0.6079E+00,         0.7376E+00, 0.8585E+00, 0.9708E+00,
         0.10745E+01, 0.11696E+01, 0.12563E+01, 0.13346E+01, 0.14045E+01, 0.14661E+01, 0.15194E+01,
         0.15645E+01, 0.16013E+01, 0.16299E+01, 0.16503E+01, 0.16626E+01, 0.16667E+01, 0.16667E+01,
         0.16667E+01, 0.16667E+01, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0,
ZD     = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.C, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0,
RP     = 0.0,
NWPI   = 30, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,  0, 0, 0,
NWING  = 1,
IFORM  = 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
PIVOT  = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
THICK  = 0.1E+01,
DIHED  = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XWI    = -0.1665E+01, -0.1665E+01, -0.11995E+02, -0.11995E+02, -0.1665E+01,   -0.1665E+01, -0.1665E+01, -0.3731E+01,
         -0.3731E+01, -0.3731E+01, -0.3731E+01, -0.3731E+01, -0.5797E+01,     -0.5797E+01, -0.5797E+01, -0.5797E+01,
         -0.5797E+01, -0.7863E+01, -0.7863E+01, -0.7863E+01, -0.7863E+01,     -0.7863E+01, -0.9929E+01, -0.9929E+01,
         -0.9929E+01, -0.9929E+01, -0.9929E+01, -0.11995E+02, -0.11995E+02,   -0.11995E+02, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,         0.0, 0.0, 0.0, 0.0, 0.0,
```

```
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,

YWI   = 0.0, 0.889E+01, 0.1405E+02, 0.1205E+02, 0.22225E+01,     0.4445E+01, 0.66675E+01,
0.241E+01, 0.4288E+01, 0.61661E+01, 0.80441E+01, 0.9922E+01,     0.482E+01, 0.63535E+01, 0.7887E+01,
0.94204E+01, 0.10954E+02, 0.723E+01, 0.8419E+01, 0.9608E+01,     0.10797E+02, 0.1198[illegible]E+02,
0.964E+01, 0.104844E+02, 0.11329E+02, 0.121735E+02, 0.13018E+02,     0.1255E+02, 0.1305E+02, 0.1355E+02,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,     0.0, 0.0, 0.0, 0.0, 0.0,
```

```
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
ZWI     = 0.0, 0.0, 0.0, 0.0, 0.1594E+00, 0.1731E+00, 0.1044E+00, 0.0,   0.1347E+00, 0.1463E+00, 0.882E-01,
          0.7E-03, 0.0, 0.11E+00, 0.1195E+00, 0.72E-01, 0.6E-03, 0.0,    0.852E-01, 0.926E-01, 0.558E-01,
          0.4E-03, 0.0, 0.605E-01, 0.658E-01, 0.397E-01, 0.3E-03,        0.358E-01, 0.389E-01, 0.235E-01,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XNACEL  = 0.0,
NACELP  = 0,
XN      = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
RN      = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
ZN      = 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
          0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,   0.0, 0.0, 0.0, 0.0, 0.0,
```

```
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,    0.0, 0.0, 0.0, 0.0, 0.0,
         0.0, 0.0, 0.0, 0.0, 0.0, 0.0,
XMACH  = 0.201E+01,
PINF   = 0.147E+02,
DADEG  = 0.0,
ARB    = 0.5E+01,
ARW    = 0.0,
ARN    = 0.0,  0.0,
$END
```

BODY PANEL CORNER POINT LOCATIONS EXPRESSED IN THE AERO COORDINATE SYSTEM
1 AND 2 INDICATE BODY PANEL LEADING-EDGE POINTS, 3 AND 4 INDICATE TRAILING-EDGE POINTS

PANEL NO	PARTS	X_1	Y_1	Z_1	X_2	Y_2	Z_2	X_3	Y_3	Z_3	X_4	Y_4	Z_4
1	1	14.70100	0.00000	1.66500	14.70100	.83250	1.44193	15.59000	0.00000	1.66500	15.59000	.83250	1.44193
2	1	15.59000	0.00000	1.66500	15.59000	.83250	1.44193	17.36800	0.00000	1.66500	17.36800	.83250	1.44193
3	1	17.36800	0.00000	1.66500	17.36800	.83250	1.44193	18.25700	0.00000	1.66500	18.25700	.83250	1.44193
4	1	18.25700	0.00000	1.66500	18.25700	.83250	1.44193	19.14600	0.00000	1.66500	19.14600	.83250	1.44193
5	1	19.14600	0.00000	1.66500	19.14600	.83250	1.44193	20.03500	0.00000	1.66500	20.03500	.83250	1.44193
6	1	20.03500	0.00000	1.66500	20.03500	.83250	1.44193	20.92400	0.00000	1.66500	20.92400	.83250	1.44193
7	1	20.92400	0.00000	1.66500	20.92400	.83250	1.44193	21.81300	0.00000	1.66500	21.81300	.83250	1.44193
8	1	21.81300	0.00000	1.66500	21.81300	.83250	1.44193	22.70200	0.00000	1.66500	22.70200	.83250	1.44193
9	1	22.70200	0.00000	1.66500	22.70200	.83250	1.44193	23.59100	0.00000	1.66500	23.59100	.83250	1.44193
10	1	23.59100	0.00000	1.66500	23.59100	.83250	1.44193	24.48000	0.00000	1.66500	24.48000	.83250	1.44193
11	1	24.48000	0.00000	1.66500	24.48000	.83250	1.44193	27.48500	0.00000	1.66500	27.48500	.83250	1.44193
12	1	27.48500	0.00000	1.66500	27.48500	.83250	1.44193	30.49000	0.00000	1.66500	30.49000	.83250	1.44193
13	1	30.49000	0.00000	1.66500	30.49000	.83250	1.44193	33.49500	0.00000	1.66500	33.49500	.83250	1.44193
14	1	33.49500	0.00000	1.66500	33.49500	.83250	1.44193	36.50000	0.00000	1.66500	36.50000	.83250	1.44193
15	1	14.70100	.83250	1.44193	14.70100	1.44193	.83250	15.59000	.83250	1.44193	15.59000	1.44193	.83250
16	1	15.59000	.83250	1.44193	15.59000	1.44193	.83250	17.36800	.83250	1.44193	17.36800	1.44193	.83250
17	1	17.36800	.83250	1.44193	17.36800	1.44193	.83250	18.25700	.83250	1.44193	18.25700	1.44193	.83250
18	1	18.25700	.83250	1.44193	18.25700	1.44193	.83250	19.14600	.83250	1.44193	19.14600	1.44193	.83250
19	1	19.14600	.83250	1.44193	19.14600	1.44193	.83250	20.03500	.83250	1.44193	20.03500	1.44193	.83250
20	1	20.03500	.83250	1.44193	20.03500	1.44193	.83250	20.92400	.83250	1.44193	20.92400	1.44193	.83250
21	1	20.92400	.83250	1.44193	20.92400	1.44193	.83250	21.81300	.83250	1.44193	21.81300	1.44193	.83250
22	1	21.81300	.83250	1.44193	21.81300	1.44193	.83250	22.70200	.83250	1.44193	22.70200	1.44193	.83250
23	1	22.70200	.83250	1.44193	22.70200	1.44193	.83250	23.59100	.83250	1.44193	23.59100	1.44193	.83250
24	1	23.59100	.83250	1.44193	23.59100	1.44193	.83250	24.48000	.83250	1.44193	24.48000	1.44193	.83250
25	1	24.48000	.83250	1.44193	24.48000	1.44193	.83250	27.48500	.83250	1.44193	27.48500	1.44193	.83250
26	1	27.48500	.83250	1.44193	27.48500	1.44193	.83250	30.49000	.83250	1.44193	30.49000	1.44193	.83250
27	1	30.49000	.83250	1.44193	30.49000	1.44193	.83250	33.49500	.83250	1.44193	33.49500	1.44193	.83250
28	1	33.49500	.83250	1.44193	33.49500	1.44193	.83250	36.50000	.83250	1.44193	36.50000	1.44193	.83250
29	1	14.70100	1.44193	.83250	14.70100	1.66500	-.00000	15.59000	1.44193	.83250	15.59000	1.66500	-.00000
30	1	15.59000	1.44193	.83250	15.59000	1.66500	-.00000	17.36800	1.44193	.83250	17.36800	1.66500	-.00000
31	1	17.36800	1.44193	.83250	17.36800	1.66500	-.00000	18.25700	1.44193	.83250	18.25700	1.66500	-.00000
32	1	18.25700	1.44193	.83250	18.25700	1.66500	-.00000	19.14600	1.44193	.83250	19.14600	1.66500	-.00000
33	1	19.14600	1.44193	.83250	19.14600	1.66500	-.00000	20.03500	1.44193	.83250	20.03500	1.66500	-.00000
34	1	20.03500	1.44193	.83250	20.03500	1.66500	-.00000	20.92400	1.44193	.83250	20.92400	1.66500	-.00000
35	1	20.92400	1.44193	.83250	20.92400	1.66500	-.00000	21.81300	1.44193	.83250	21.81300	1.66500	-.00000
36	1	21.81300	1.44193	.83250	21.81300	1.66500	-.00000	22.70200	1.44193	.83250	22.70200	1.66500	-.00000
37	1	22.70200	1.44193	.83250	22.70200	1.66500	-.00000	23.59100	1.44193	.83250	23.59100	1.66500	-.00000
38	1	23.59100	1.44193	.83250	23.59100	1.66500	-.00000	24.48000	1.44193	.83250	24.48000	1.66500	-.00000
39	1	24.48000	1.44193	.83250	24.48000	1.66500	-.00000	27.48500	1.44193	.83250	27.48500	1.66500	-.00000
40	1	27.48500	1.44193	.83250	27.48500	1.66500	-.00000	30.49000	1.44193	.83250	30.49000	1.66500	-.00000
41	1	30.49000	1.44193	.83250	30.49000	1.66500	-.00000	33.49500	1.44193	.83250	33.49500	1.66500	-.00000
42	1	33.49500	1.44193	.83250	33.49500	1.66500	-.00000	36.50000	1.44193	.83250	36.50000	1.66500	-.00000
43	1	14.70100	1.66500	-.00000	14.70100	1.44193	-.83250	15.59000	1.66500	-.00000	15.59000	1.44193	-.83250
44	1	15.59000	1.66500	-.00000	15.59000	1.44193	-.83250	17.36800	1.66500	-.00000	17.36800	1.44193	-.83250
45	1	17.36800	1.66500	-.00000	17.36800	1.44193	-.83250	18.25700	1.66500	-.00000	18.25700	1.44193	-.83250
46	1	18.25700	1.66500	-.00000	18.25700	1.44193	-.83250	19.14600	1.66500	-.00000	19.14600	1.44193	-.83250
47	1	19.14600	1.66500	-.00000	19.14600	1.44193	-.83250	20.03500	1.66500	-.00000	20.03500	1.44193	-.83250
48	1	20.03500	1.66500	-.00000	20.03500	1.44193	-.83250	20.92400	1.66500	-.00000	20.92400	1.44193	-.83250
49	1	20.92400	1.66500	-.00000	20.92400	1.44193	-.83250	21.81300	1.66500	-.00000	21.81300	1.44193	-.83250
50	1	21.81300	1.66500	-.00000	21.81300	1.44193	-.83250	22.70200	1.66500	-.00000	22.70200	1.44193	-.83250
51	1	22.70200	1.66500	-.00000	22.70200	1.44193	-.83250	23.59100	1.66500	-.00000	23.59100	1.44193	-.83250
52	1	23.59100	1.66500	-.00000	23.59100	1.44193	-.83250	24.48000	1.66500	-.00000	24.48000	1.44193	-.83250
53	1	24.48000	1.66500	-.00000	24.48000	1.44193	-.83250	27.48500	1.66500	-.00000	27.48500	1.44193	-.83250
54	1	27.48500	1.66500	-.00000	27.48500	1.44193	-.83250	30.49000	1.66500	-.00000	30.49000	1.44193	-.83250
55	1	30.49000	1.66500	-.00000	30.49000	1.44193	-.83250	33.49500	1.66500	-.00000	33.49500	1.44193	-.83250
56	1	33.49500	1.66500	-.00000	33.49500	1.44193	-.83250	36.50000	1.66500	-.00000	36.50000	1.44193	-.83250
57	1	14.70100	1.44193	-.83250	14.70100	.83250	-1.44193	15.59000	1.44193	-.83250	15.59000	.83250	-1.44193
58	1	15.59000	1.44193	-.83250	15.59000	.83250	-1.44193	17.36800	1.44193	-.83250	17.36800	.83250	-1.44193

59	1	17.36800	1.44193	-.83250	17.36800	.83250	-1.44193	18.25700	1.44193	-.83250	18.25700	.83250	-1.44193
60	1	18.25700	1.44193	-.83250	18.25700	.83250	-1.44193	19.14600	1.44193	-.83250	19.14600	.83250	-1.44193
61	1	19.14600	1.44193	-.83250	19.14600	.83250	-1.44193	20.03500	1.44193	-.83250	20.03500	.83250	-1.44193
62	1	20.03500	1.44193	-.83250	20.03500	.83250	-1.44193	20.92400	1.44193	-.83250	20.92400	.83250	-1.44193
63	1	20.92400	1.44193	-.83250	20.92400	.83250	-1.44193	21.81300	1.44193	-.83250	21.81300	.83250	-1.44193
64	1	21.81300	1.44193	-.83250	21.81300	.83250	-1.44193	22.70200	1.44193	-.83250	22.70200	.83250	-1.44193
65	1	22.70200	1.44193	-.83250	22.70200	.83250	-1.44193	23.59100	1.44193	-.83250	23.59100	.83250	-1.44193
66	1	23.59100	1.44193	-.83250	23.59100	.83250	-1.44193	24.48000	1.44193	-.83250	24.48000	.83250	-1.44193
67	1	24.48000	1.44193	-.83250	24.48000	.83250	-1.44193	27.48500	1.44193	-.83250	27.48500	.83250	-1.44193
68	1	27.48500	1.44193	-.83250	27.48500	.83250	-1.44193	30.49000	1.44193	-.83250	30.49000	.83250	-1.44193
69	1	30.49000	1.44193	-.83250	30.49000	.83250	-1.44193	33.49500	1.44193	-.83250	33.49500	.83250	-1.44193
70	1	33.49500	1.44193	-.83250	33.49500	.83250	-1.44193	36.50000	1.44193	-.83250	36.50000	.83250	-1.44193
71	1	14.70100	.83250	-1.44193	14.70100	-.00000	-1.66500	15.59000	.83250	-1.44193	15.59000	-.00000	-1.66500
72	1	15.59000	.83250	-1.44193	15.59000	-.00000	-1.66500	17.36800	.83250	-1.44193	17.36800	-.00000	-1.66500
73	1	17.36800	.83250	-1.44193	17.36800	-.00000	-1.66500	18.25700	.83250	-1.44193	18.25700	-.00000	-1.66500
74	1	18.25700	.83250	-1.44193	18.25700	-.00000	-1.66500	19.14600	.83250	-1.44193	19.14600	-.00000	-1.66500
75	1	19.14600	.83250	-1.44193	19.14600	-.00000	-1.66500	20.03500	.83250	-1.44193	20.03500	-.00000	-1.66500
76	1	20.03500	.83250	-1.44193	20.03500	-.00000	-1.66500	20.92400	.83250	-1.44193	20.92400	-.00000	-1.66500
77	1	20.92400	.83250	-1.44193	20.92400	-.00000	-1.66500	21.81300	.83250	-1.44193	21.81300	-.00000	-1.66500
78	1	21.81300	.83250	-1.44193	21.81300	-.00000	-1.66500	22.70200	.83250	-1.44193	22.70200	-.00000	-1.66500
79	1	22.70200	.83250	-1.44193	22.70200	-.00000	-1.66500	23.59100	.83250	-1.44193	23.59100	-.00000	-1.66500
80	1	23.59100	.83250	-1.44193	23.59100	-.00000	-1.66500	24.48000	.83250	-1.44193	24.48000	-.00000	-1.66500
81	1	24.48000	.83250	-1.44193	24.48000	-.00000	-1.66500	27.48500	.83250	-1.44193	27.48500	-.00000	-1.66500
82	1	27.48500	.83250	-1.44193	27.48500	-.00000	-1.66500	30.49000	.83250	-1.44193	30.49000	-.00000	-1.66500
83	1	30.49000	.83250	-1.44193	30.49000	-.00000	-1.66500	33.49500	.83250	-1.44193	33.49500	-.00000	-1.66500
84	1	33.49500	.83250	-1.44193	33.49500	-.00000	-1.66500	36.50000	.83250	-1.44193	36.50000	-.00000	-1.66500

BODY PANEL CENTROID AND CONTROL POINT LOCATIONS EXPRESSED IN THE AERO COORDINATE SYSTEM

PANEL	X_C	Y_C	Z_C	X_{CP}	Y_{CP}	Z_{CP}	AREA	THETA-INCLIN	ALPHA-INCID	CHORD
1	15.14550	.41625	1.55347	15.54555	.41625	1.55347	.76620	-.26180	0.00000	.88900
2	16.47900	.41625	1.55347	17.27910	.41625	1.55347	1.53240	-.26180	0.00000	1.77800
3	17.81250	.41625	1.55347	18.21255	.41625	1.55347	.76620	-.26180	0.00000	.88900
4	18.70150	.41625	1.55347	19.10155	.41625	1.55347	.76620	-.26180	0.00000	.88900
5	19.59050	.41625	1.55347	19.99055	.41625	1.55347	.76620	-.26180	0.00000	.88900
6	20.47950	.41625	1.55347	20.87955	.41625	1.55347	.76620	-.26180	0.00000	.88900
7	21.36850	.41625	1.55347	21.76855	.41625	1.55347	.76620	-.26180	0.00000	.88900
8	22.25750	.41625	1.55347	22.65755	.41625	1.55347	.76620	-.26180	0.00000	.88900
9	23.14650	.41625	1.55347	23.54655	.41625	1.55347	.76620	-.26180	0.00000	.88900
10	24.03550	.41625	1.55347	24.43555	.41625	1.55347	.76620	-.26180	0.00000	.88900
11	25.98250	.41625	1.55347	27.33475	.41625	1.55347	2.58991	-.26180	0.00000	3.00500
12	28.98750	.41625	1.55347	30.33975	.41625	1.55347	2.58991	-.26180	0.00000	3.00500
13	31.99250	.41625	1.55347	33.34475	.41625	1.55347	2.58991	-.26180	0.00000	3.00500
14	34.99750	.41625	1.55347	36.34975	.41625	1.55347	2.58991	-.26180	0.00000	3.00500
15	15.14550	1.13722	1.13722	15.54555	1.13722	1.13722	.76620	-.78540	0.00000	.88900
16	16.47900	1.13722	1.13722	17.27910	1.13722	1.13722	1.53240	-.78540	0.00000	1.77800
17	17.81250	1.13722	1.13722	18.21255	1.13722	1.13722	.76620	-.78540	0.00000	.88900
18	18.70150	1.13722	1.13722	19.10155	1.13722	1.13722	.76620	-.78540	0.00000	.88900
19	19.59050	1.13722	1.13722	19.99055	1.13722	1.13722	.76620	-.78540	0.00000	.88900
20	20.47950	1.13722	1.13722	20.87955	1.13722	1.13722	.76620	-.78540	0.00000	.88900
21	21.36850	1.13722	1.13722	21.76855	1.13722	1.13722	.76620	-.78540	0.00000	.88900
22	22.25750	1.13722	1.13722	22.65755	1.13722	1.13722	.76620	-.78540	0.00000	.88900
23	23.14650	1.13722	1.13722	23.54655	1.13722	1.13722	.76620	-.78540	0.00000	.88900
24	24.03550	1.13722	1.13722	24.43555	1.13722	1.13722	.76620	-.78540	0.00000	.88900
25	25.98250	1.13722	1.13722	27.33475	1.13722	1.13722	2.58991	-.78540	0.00000	3.00500
26	28.98750	1.13722	1.13722	30.33975	1.13722	1.13722	2.58991	-.78540	0.00000	3.00500
27	31.99250	1.13722	1.13722	33.34475	1.13722	1.13722	2.58991	-.78540	0.00000	3.00500
28	34.99750	1.13722	1.13722	36.34975	1.13722	1.13722	2.58991	-.78540	0.00000	3.00500
29	15.14550	1.55347	.41625	15.54555	1.55347	.41625	.76620	-1.30900	0.00000	.88900
30	16.47900	1.55347	.41625	17.27910	1.55347	.41625	1.53240	-1.30900	0.00000	1.77800
31	17.81250	1.55347	.41625	18.21255	1.55347	.41625	.76620	-1.30900	0.00000	.88900
32	18.70150	1.55347	.41625	19.10155	1.55347	.41625	.76620	-1.30900	0.00000	.88900
33	19.59050	1.55347	.41625	19.99055	1.55347	.41625	.76620	-1.30900	0.00000	.88900
34	20.47950	1.55347	.41625	20.87955	1.55347	.41625	.76620	-1.30900	0.00000	.88900
35	21.36850	1.55347	.41625	21.76855	1.55347	.41625	.76620	-1.30900	0.00000	.88900
36	22.25750	1.55347	.41625	22.65755	1.55347	.41625	.76620	-1.30900	0.00000	.88900
37	23.14650	1.55347	.41625	23.54655	1.55347	.41625	.76620	-1.30900	0.00000	.88900
38	24.03550	1.55347	.41625	24.43555	1.55347	.41625	.76620	-1.30900	0.00000	.88900
39	25.98250	1.55347	.41625	27.33475	1.55347	.41625	2.58991	-1.30900	0.00000	3.00500
40	28.98750	1.55347	.41625	30.33975	1.55347	.41625	2.58991	-1.30900	0.00000	3.00500
41	31.99250	1.55347	.41625	33.34475	1.55347	.41625	2.58991	-1.30900	0.00000	3.00500
42	34.99750	1.55347	.41625	36.34975	1.55347	.41625	2.58991	-1.30900	0.00000	3.00500
43	15.14550	1.55347	-.41625	15.54555	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
44	16.47900	1.55347	-.41625	17.27910	1.55347	-.41625	1.53240	-1.83260	0.00000	1.77800
45	17.81250	1.55347	-.41625	18.21255	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
46	18.70150	1.55347	-.41625	19.10155	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
47	19.59050	1.55347	-.41625	19.99055	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
48	20.47950	1.55347	-.41625	20.87955	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
49	21.36850	1.55347	-.41625	21.76855	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
50	22.25750	1.55347	-.41625	22.65755	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
51	23.14650	1.55347	-.41625	23.54655	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
52	24.03550	1.55347	-.41625	24.43555	1.55347	-.41625	.76620	-1.83260	0.00000	.88900
53	25.98250	1.55347	-.41625	27.33475	1.55347	-.41625	2.58991	-1.83260	0.00000	3.00500
54	28.98750	1.55347	-.41625	30.33975	1.55347	-.41625	2.58991	-1.83260	0.00000	3.00500
55	31.99250	1.55347	-.41625	33.34475	1.55347	-.41625	2.58991	-1.83260	0.00000	3.00500
56	34.99750	1.55347	-.41625	36.34975	1.55347	-.41625	2.58991	-1.83260	0.00000	3.00500
57	15.14550	1.13722	-1.13722	15.54555	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
58	16.47900	1.13722	-1.13722	17.27910	1.13722	-1.13722	1.53240	-2.35619	0.00000	1.77800
59	17.81250	1.13722	-1.13722	18.21255	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900

60	18.70150	1.13722	-1.13722	19.10155	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
61	19.59050	1.13722	-1.13722	19.99055	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
62	20.47950	1.13722	-1.13722	20.87955	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
63	21.36850	1.13722	-1.13722	21.76855	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
64	22.25750	1.13722	-1.13722	22.65755	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
65	23.14650	1.13722	-1.13722	23.54655	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
66	24.03550	1.13722	-1.13722	24.43555	1.13722	-1.13722	.76620	-2.35619	0.00000	.88900
67	25.98250	1.13722	-1.13722	27.33475	1.13722	-1.13722	2.58991	-2.35619	0.00000	3.00500
68	28.98750	1.13722	-1.13722	30.33975	1.13722	-1.13722	2.58991	-2.35619	0.00000	3.00500
69	31.99250	1.13722	-1.13722	33.34475	1.13722	-1.13722	2.58991	-2.35619	0.00000	3.00500
70	34.99750	1.13722	-1.13722	36.34975	1.13722	-1.13722	2.58991	-2.35619	0.00000	3.00500
71	15.14550	.41625	-1.55347	15.54555	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
72	16.47900	.41625	-1.55347	17.27910	.41625	-1.55347	1.53240	-2.87979	0.00000	1.77800
73	17.81250	.41625	-1.55347	18.21255	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
74	18.70150	.41625	-1.55347	19.10155	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
75	19.59050	.41625	-1.55347	19.99055	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
76	20.47950	.41625	-1.55347	20.87955	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
77	21.36850	.41625	-1.55347	21.76855	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
78	22.25750	.41625	-1.55347	22.65755	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
79	23.14650	.41625	-1.55347	23.54655	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
80	24.03550	.41625	-1.55347	24.43555	.41625	-1.55347	.76620	-2.87979	0.00000	.88900
81	25.98250	.41625	-1.55347	27.33475	.41625	-1.55347	2.58991	-2.87979	0.0000C	3.00500
82	28.98750	.41625	-1.55347	30.33975	.41625	-1.55347	2.58991	-2.87979	0.00000	3.00500
83	31.99250	.41625	-1.55347	33.34475	.41625	-1.55347	2.58991	-2.87979	0.00000	3.00500
84	34.99750	.41625	-1.55347	36.34975	.41625	-1.55347	2.58991	-2.87979	0.00000	3.00500

WING PANEL CORNER POINT LOCATIONS EXPRESSED IN THE AERO COORDINATE SYSTEM
1 AND 2 INDICATE WING PANEL LEADING-EDGE POINTS, 3 AND 4 INDICATE TRAILING-EDGE POINTS

WING NO. 1   YPIVOT = 1.665   ZPIVOT = 0.000   WING DIHEDRAL = 0.000 RADIANS

PANEL NO	PARTS	X_1	Y_1	Z_1	X_2	Y_2	Z_2	X_3	Y_3	Z_3	X_4	Y_4	Z_4
1	1	15.59000	1.66500	0.00000	18.00000	3.73100	0.00000	16.47900	1.66500	0.00000	18.75120	3.73100	0.00000
2	1	16.47900	1.66500	0.00000	18.75120	3.73100	0.00000	17.36800	1.66500	0.00000	19.50240	3.73100	0.00000
3	1	17.36800	1.66500	0.00000	19.50240	3.73100	0.00000	18.25700	1.66500	0.00000	20.25360	3.73100	0.00000
4	1	18.25700	1.66500	0.00000	20.25360	3.73100	0.00000	19.14600	1.66500	0.00000	21.00480	3.73100	0.00000
5	1	19.14600	1.66500	0.00000	21.00480	3.73100	0.00000	20.03500	1.66500	0.00000	21.75600	3.73100	0.00000
6	1	20.03500	1.66500	0.00000	21.75600	3.73100	0.00000	20.92400	1.66500	0.00000	22.50720	3.73100	0.00000
7	1	20.92400	1.66500	0.00000	22.50720	3.73100	0.00000	21.81300	1.66500	0.00000	23.25840	3.73100	0.00000
8	1	21.81300	1.66500	0.00000	23.25840	3.73100	0.00000	22.70200	1.66500	0.00000	24.00960	3.73100	0.00000
9	1	22.70200	1.66500	0.00000	24.00960	3.73100	0.00000	23.59100	1.66500	0.00000	24.76080	3.73100	0.00000
10	1	23.59100	1.66500	0.00000	24.76080	3.73100	0.00000	24.48000	1.66500	0.00000	25.51200	3.73100	0.00000
11	1	18.00000	3.73100	0.00000	20.41000	5.79700	0.00000	18.75120	3.73100	0.00000	21.02340	5.79700	0.00000
12	1	18.75120	3.73100	0.00000	21.02340	5.79700	0.00000	19.50240	3.73100	0.00000	21.63680	5.79700	0.00000
13	1	19.50240	3.73100	0.00000	21.63680	5.79700	0.00000	20.25360	3.73100	0.00000	22.25020	5.79700	0.00000
14	1	20.25360	3.73100	0.00000	22.25020	5.79700	0.00000	21.00480	3.73100	0.00000	22.86360	5.79700	0.00000
15	1	21.00480	3.73100	0.00000	22.86360	5.79700	0.00000	21.75600	3.73100	0.00000	23.47700	5.79700	0.00000
16	1	21.75600	3.73100	0.00000	23.47700	5.79700	0.00000	22.50720	3.73100	0.00000	24.09040	5.79700	0.00000
17	1	22.50720	3.73100	0.00000	24.09040	5.79700	0.00000	23.25840	3.73100	0.00000	24.70380	5.79700	0.00000
18	1	23.25840	3.73100	0.00000	24.70380	5.79700	0.00000	24.00960	3.73100	0.00000	25.31720	5.79700	0.00000
19	1	24.00960	3.73100	0.00000	25.31720	5.79700	0.00000	24.76080	3.73100	0.00000	25.93060	5.79700	0.00000
20	1	24.76080	3.73100	0.00000	25.93060	5.79700	0.00000	25.51200	3.73100	0.00000	26.54400	5.79700	0.00000
21	1	20.41000	5.79700	0.00000	22.82000	7.86300	0.00000	21.02340	5.79700	0.00000	23.29560	7.86300	0.00000
22	1	21.02340	5.79700	0.00000	23.29560	7.86300	0.00000	21.63680	5.79700	0.00000	23.77120	7.86300	0.00000
23	1	21.63680	5.79700	0.00000	23.77120	7.86300	0.00000	22.25020	5.79700	0.00000	24.24680	7.86300	0.00000
24	1	22.25020	5.79700	0.00000	24.24680	7.86300	0.00000	22.86360	5.79700	0.00000	24.72240	7.86300	0.00000
25	1	22.86360	5.79700	0.00000	24.72240	7.86300	0.00000	23.47700	5.79700	0.00000	25.19800	7.86300	0.00000
26	1	23.47700	5.79700	0.00000	25.19800	7.86300	0.00000	24.09040	5.79700	0.00000	25.67360	7.86300	0.00000
27	1	24.09040	5.79700	0.00000	25.67360	7.86300	0.00000	24.70380	5.79700	0.00000	26.14920	7.86300	0.00000
28	1	24.70380	5.79700	0.00000	26.14920	7.86300	0.00000	25.31720	5.79700	0.00000	26.62480	7.86300	0.00000
29	1	25.31720	5.79700	0.00000	26.62480	7.86300	0.00000	25.93060	5.79700	0.00000	27.10040	7.86300	0.00000
30	1	25.93060	5.79700	0.00000	27.10040	7.86300	0.00000	26.54400	5.79700	0.00000	27.57600	7.86300	0.00000
31	1	22.82000	7.86300	0.00000	25.23000	9.92900	0.00000	23.29560	7.86300	0.00000	25.56780	9.92900	0.00000
32	1	23.29560	7.86300	0.00000	25.56780	9.92900	0.00000	23.77120	7.86300	0.00000	25.90560	9.92900	0.00000
33	1	23.77120	7.86300	0.00000	25.90560	9.92900	0.00000	24.24680	7.86300	0.00000	26.24340	9.92900	0.00000
34	1	24.24680	7.86300	0.00000	26.24340	9.92900	0.00000	24.72240	7.86300	0.00000	26.58120	9.92900	0.00000
35	1	24.72240	7.86300	0.00000	26.58120	9.92900	0.00000	25.19800	7.86300	0.00000	26.91900	9.92900	0.00000
36	1	25.19800	7.86300	0.00000	26.91900	9.92900	0.00000	25.67360	7.86300	0.00000	27.25680	9.92900	0.00000
37	1	25.67360	7.86300	0.00000	27.25680	9.92900	0.00000	26.14920	7.86300	0.00000	27.59460	9.92900	0.00000
38	1	26.14920	7.86300	0.00000	27.59460	9.92900	0.00000	26.62480	7.86300	0.00000	27.93240	9.92900	0.00000
39	1	26.62480	7.86300	0.00000	27.93240	9.92900	0.00000	27.10040	7.86300	0.00000	28.27020	9.92900	0.00000
40	1	27.10040	7.86300	0.00000	28.27020	9.92900	0.00000	27.57600	7.86300	0.00000	28.60800	9.92900	0.00000
41	1	25.23000	9.92900	0.00000	27.64000	11.99500	0.00000	25.56780	9.92900	0.00000	27.84000	11.99500	0.00000
42	1	25.56780	9.92900	0.00000	27.84000	11.99500	0.00000	25.90560	9.92900	0.00000	28.04000	11.99500	0.00000
43	1	25.90560	9.92900	0.00000	28.04000	11.99500	0.00000	26.24340	9.92900	0.00000	28.24000	11.99500	0.00000
44	1	26.24340	9.92900	0.00000	28.24000	11.99500	0.00000	26.58120	9.92900	0.00000	28.44000	11.99500	0.00000
45	1	26.58120	9.92900	0.00000	28.44000	11.99500	0.00000	26.91900	9.92900	0.00000	28.64000	11.99500	0.00000
46	1	26.91900	9.92900	0.00000	28.64000	11.99500	0.00000	27.25680	9.92900	0.00000	28.84000	11.99500	0.00000
47	1	27.25680	9.92900	0.00000	28.84000	11.99500	0.00000	27.59460	9.92900	0.00000	29.04000	11.99500	0.00000
48	1	27.59460	9.92900	0.00000	29.04000	11.99500	0.00000	27.93240	9.92900	0.00000	29.24000	11.99500	0.00000
49	1	27.93240	9.92900	0.00000	29.24000	11.99500	0.00000	28.27020	9.92900	0.00000	29.44000	11.99500	0.00000
50	1	28.27020	9.92900	0.00000	29.44000	11.99500	0.00000	28.60800	9.92900	0.00000	29.64000	11.99500	0.00000

WING PANEL CENTROID AND CONTROL POINT LOCATIONS EXPRESSED IN THE AERO COORDINATE SYSTEM

PANEL	X_C	Y_C	Z_C	X_{CP}	Y_{CP}	Z_{CP}	AREA	THETA-DIHED	ALPHA-CAMBER	CHORD
1	17.17227	2.66907	0.00000	17.54218	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
2	17.99430	2.66907	0.00000	18.36421	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
3	18.81633	2.66907	0.00000	19.18624	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
4	19.63836	2.66907	0.00000	20.00827	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
5	20.46039	2.66907	0.00000	20.83030	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
6	21.28242	2.66907	0.00000	21.65233	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
7	22.10445	2.66907	0.00000	22.47436	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
8	22.92648	2.66907	0.00000	23.29639	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
9	23.74851	2.66907	0.00000	24.11842	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
10	24.57053	2.66907	0.00000	24.94045	2.66907	0.00000	1.69433	0.00000	0.00000	.82203
11	19.50675	4.72923	0.00000	19.81483	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
12	20.19137	4.72923	0.00000	20.49945	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
13	20.87599	4.72923	0.00000	21.18407	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
14	21.56061	4.72923	0.00000	21.86868	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
15	22.24523	4.72923	0.00000	22.55330	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
16	22.92984	4.72923	0.00000	23.23792	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
17	23.61446	4.72923	0.00000	23.92254	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
18	24.29908	4.72923	0.00000	24.60716	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
19	24.98370	4.72923	0.00000	25.29178	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
20	25.66832	4.72923	0.00000	25.97640	4.72923	0.00000	1.40963	0.00000	0.00000	.68462
21	21.83788	6.78643	0.00000	22.08421	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
22	22.38528	6.78643	0.00000	22.63162	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
23	22.93269	6.78643	0.00000	23.17902	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
24	23.48010	6.78643	0.00000	23.72643	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
25	24.02750	6.78643	0.00000	24.27383	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
26	24.57491	6.78643	0.00000	24.82124	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
27	25.12231	6.78643	0.00000	25.36865	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
28	25.66972	6.78643	0.00000	25.91605	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
29	26.21713	6.78643	0.00000	26.46346	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
30	26.76453	6.78643	0.00000	27.01087	6.78643	0.00000	1.12494	0.00000	0.00000	.54741
31	24.16225	8.83767	0.00000	24.34701	8.83767	0.00000	.84024	0.00000	0.00000	.41059
32	24.57284	8.83767	0.00000	24.75760	8.83767	0.00000	.84024	0.00000	0.00000	.41059
33	24.98343	8.83767	0.00000	25.16820	8.83767	0.00000	.84024	0.00000	0.00000	.41059
34	25.39402	8.83767	0.00000	25.57879	8.83767	0.00000	.84024	0.00000	0.00000	.41059
35	25.80461	8.83767	0.00000	25.98938	8.83767	0.00000	.84024	0.00000	0.00000	.41059
36	26.21520	8.83767	0.00000	26.39997	8.83767	0.00000	.84024	0.00000	0.00000	.41059
37	26.62579	8.83767	0.00000	26.81056	8.83767	0.00000	.84024	0.00000	0.00000	.41059
38	27.03638	8.83767	0.00000	27.22115	8.83767	0.00000	.84024	0.00000	0.00000	.41059
39	27.44697	8.83767	0.00000	27.63174	8.83767	0.00000	.84024	0.00000	0.00000	.41059
40	27.85757	8.83767	0.00000	28.04233	8.83767	0.00000	.84024	0.00000	0.00000	.41059
41	26.46947	10.87377	0.00000	26.59313	10.87377	0.00000	.55555	0.00000	0.00000	.27478
42	26.74426	10.87377	0.00000	26.86791	10.87377	0.00000	.55555	0.00000	0.00000	.27478
43	27.01904	10.87377	0.00000	27.14270	10.87377	0.00000	.55555	0.00000	0.00000	.27478
44	27.29383	10.87377	0.00000	27.41748	10.87377	0.00000	.55555	0.00000	0.00000	.27478
45	27.56861	10.87377	0.00000	27.69227	10.87377	0.00000	.55555	0.00000	0.00000	.27478
46	27.84340	10.87377	0.00000	27.96705	10.87377	0.00000	.55555	0.00000	0.00000	.27478
47	28.11818	10.87377	0.00000	28.24184	10.87377	0.00000	.55555	0.00000	0.00000	.27478
48	28.39297	10.87377	0.00000	28.51662	10.87377	0.00000	.55555	0.00000	0.00000	.27478
49	28.66775	10.87377	0.00000	28.79140	10.87377	0.00000	.55555	0.00000	0.00000	.27478
50	28.94254	10.87377	0.00000	29.06619	10.87377	0.00000	.55555	0.00000	0.00000	.27478

EXIT INTAPE

EXIT EVAL

EXIT EVAL

```
DESCRIPTION OF CASE REQUESTED

SYMMETRICAL CONFIGURATION - PANELS LOCATED ON BOTH SIDES OF X-Z PLANE(SYM = 1.)

CASE = 2.          CALCULATE CL, GIVEN SHAPE

CPCALC = 0.        LINEAR CP

POLAR = 1.         POLARS REQUESTED

THICK = 1.         WING THICKNESS PRESSURES TO BE ADDED

VOUT = 0.          VELOCITY COMPONENTS NOT TO BE PRINTED

MACH NUMBER =    2.0100

POINT ABOUT WHICH THE MOMENTS ARE TO BE COMPUTED
 X-COORDINATE =    0.0000
 Z-COORDINATE =    0.0000

REFERENCE CHORD LENGTH =    1.0000

WING REFERENCE AREA =   56.2468

WING SEMI-SPAN =     1.0000

HEIGHT OF WING PLANE ABOVE BODY AXIS =     0.0000

INCLINATION OF BODY AXIS WITH RESPECT TO DEFINING AXIS =     0.0000  DEG.

ANGLE OF ATTACK WITH RESPECT TO BODY AXIS =     5.0000  DEG.
```

PRESSURES, FORCES, AND MOMENTS ON ISOLATED BODY

CD = .00985 CL = .02033 CM = -.11461

BODY PRESSURE COEFFICIENTS(CP)

THETA(DEG.) X	0.0000	30.0000	60.0000	90.0000	120.0000	150.0000	180.0000
0.0000	.12638	.13607	.16616	.21578	.27524	.32501	.34454
.7300	.11196	.12129	.15040	.19876	.25703	.30596	.32520
1.4600	.08963	.09797	.12453	.16979	.22539	.27265	.29134
2.1900	.08156	.08911	.11368	.15656	.21022	.25633	.27465
2.9200	.06710	.07400	.09689	.13771	.18955	.23448	.25241
3.6500	.05376	.05992	.08089	.11935	.16914	.21278	.23027
4.3800	.04049	.04587	.06482	.10079	.14842	.19067	.20769
5.1100	.02795	.03250	.04931	.08266	.12800	.16878	.18531
5.8400	.01576	.01944	.03401	.06459	.10749	.14671	.16272
6.5700	.00414	.00691	.01912	.04677	.08710	.12467	.14011
7.3000	-.01976	-.01824	-.00931	.01422	.05081	.08588	.10047
9.2467	-.04166	-.04303	-.04159	-.02744	.00080	.03041	.04313
11.1933	-.04718	-.05153	-.05788	-.05347	-.03395	-.01008	.00068
13.1400	-.02317	-.02981	-.04214	-.04519	-.03229	-.01275	-.00346
15.0867	-.00714	-.01505	-.03072	-.03808	-.02921	-.01244	-.00413
17.0333	.00291	-.00554	-.02270	-.03210	-.02531	-.01006	-.00231
18.9800	.00802	-.00052	-.01798	-.02795	-.02190	-.00730	.00019
20.9267	.01049	.00208	-.01512	-.02491	-.01887	-.00442	.00299
22.8733	.01146	.00324	-.01348	-.02279	-.01645	-.00189	.00553
24.8200	.01175	.00372	-.01254	-.02133	-.01462	.00011	.00759
26.7667	.01184	.00395	-.01195	-.02032	-.01330	.00161	.00913
28.7133	.01195	.00417	-.01148	-.01956	-.01230	.00274	.01030
30.6600	.01213	.00441	-.01108	-.01899	-.01159	.00354	.01112
32.6067	.01236	.00467	-.01074	-.01855	-.01107	.00410	.01170
34.5533	.01263	.00495	-.01043	-.01821	-.01071	.00447	.01207
36.5000	.01329	.00562	-.00973	-.01747	-.00993	.00526	.01287

INCREMENTAL PRESSURES, FORCES, AND MOMENTS ON BODY PANELS DUE TO WING

CD = .00391   CL = .04487   CM = -1.08710

BODY PANEL PRESSURE COEFFICIENT(CP)

THETA(DEG.)	15.0000	45.0000	75.0000	105.0000	135.0000	165.0000
ROW NO.						
1	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000	-0.00000
2	-.00043	-.00167	-.08854	.10290	.00194	.00049
3	.00010	-.03029	-.10842	.12602	.03531	-.00013
4	-.00323	-.05772	-.09385	.11178	.06753	.00371
5	-.01864	-.06848	-.08109	.09886	.08084	.02180
6	-.03965	-.06519	-.07267	.08979	.07846	.04632
7	-.05621	-.05866	-.06123	.07839	.07198	.06682
8	-.06354	-.05818	-.05944	.07266	.07282	.07624
9	-.06840	-.05575	-.06076	.06500	.06955	.08325
10	-.06681	-.06076	-.06991	.06527	.06918	.08299
11	-.07271	-.05738	-.03344	.00491	.04445	.07658
12	-.05496	-.02850	-.00645	.00591	.02307	.03984
13	-.01835	-.00944	-.00372	.00713	.01996	.01979
14	-.00382	-.00108	-.00320	-.00215	.00438	.00843

BODY PANEL SLOPE(DR/DX)

THETA(DEG.)	15.0000	45.0000	75.0000	105.0000	135.0000	165.0000
ROW NO.						
1	-.08521	-.06262	-.02350	.02167	.06079	.08338
2	-.08509	-.06251	-.02339	.02178	.06090	.08349
3	-.08482	-.06224	-.02312	.02206	.06118	.08376
4	-.08458	-.06199	-.02287	.02230	.06142	.08401
5	-.08439	-.06180	-.02268	.02249	.06161	.08420
6	-.08420	-.06161	-.02249	.02268	.06180	.08439
7	-.08408	-.06149	-.02237	.02280	.06192	.08451
8	-.08396	-.06138	-.02226	.02292	.06204	.08462
9	-.08390	-.06131	-.02219	.02298	.06210	.08469
10	-.08385	-.06126	-.02214	.02303	.06215	.08473
11	-.08396	-.06138	-.02226	.02292	.06204	.08462
12	-.08416	-.06158	-.02246	.02271	.06183	.08442
13	-.08437	-.06178	-.02266	.02251	.06163	.08422
14	-.08459	-.06201	-.02289	.02229	.06141	.08399

PRESSURES, FORCES, AND MOMENTS ON WING PANELS IN PRESENCE OF BODY

CD = .02268 CL = .25441 CM = -5.93827

SECTION CD DISTRIBUTION

SPANWISE STATION	1	2	3	4	5
	.01821	.02259	.02586	.02629	.02466

SECTION CL DISTRIBUTION

SPANWISE STATION	1	2	3	4	5
	.20894	.24837	.28822	.29282	.28191

UPPER SURFACE WING PANEL PRESSURE COEFFICIENTS(CP)

SPANWISE STATION CHORDWISE STATION	1	2	3	4	5
1	-.18718	-.15204	-.18585	-.19518	-.14959
2	-.15988	-.17569	-.17826	-.18388	-.15129
3	-.12633	-.18300	-.17832	-.17304	-.16301
4	-.11276	-.17666	-.18335	-.17651	-.16914
5	-.08966	-.15922	-.15723	-.17379	-.16907
6	-.08231	-.14686	-.12687	-.16909	-.16889
7	-.08209	-.12585	-.13803	-.16496	-.16705
8	-.07656	-.07986	-.14025	-.15503	-.16503
9	-.07950	-.03800	-.13214	-.14806	-.16475
10	-.07877	-.04064	-.12008	-.14484	-.16630

LOWER SURFACE WING PANEL PRESSURE COEFFICIENTS(CP)

SPANWISE STATION CHORDWISE STATION	1	2	3	4	5
1	.20593	.16504	.10942	.09080	.13103
2	.16103	.15584	.11964	.10271	.12951
3	.12764	.14942	.12413	.11481	.11827
4	.11462	.12657	.12540	.11275	.11271
5	.09035	.10926	.15681	.11733	.11337
6	.07812	.09359	.18636	.12458	.11415
7	.07302	.07873	.15989	.13181	.11664
8	.05902	.09938	.13703	.14529	.11945
9	.05905	.12564	.12259	.15548	.12059
10	.05353	.11186	.11156	.15948	.12001

WING PANEL PRESSURE DIFFERENCE(CL)

SPANWISE STATION CHORDWISE STATION	1	2	3	4	5
1	.39311	.31708	.29527	.28598	.28062
2	.32091	.33153	.29791	.28659	.28081
3	.25397	.33241	.30245	.28785	.28127
4	.22738	.30323	.30875	.28926	.28185
5	.18001	.26848	.31403	.29111	.28244
6	.16043	.24046	.31323	.29367	.28303
7	.15511	.20457	.29792	.29677	.28369
8	.13558	.17924	.27727	.30032	.28447
9	.13855	.16364	.25473	.30354	.28534
10	.13230	.15250	.23163	.30433	.28631

UPPER SURFACE WING PANEL SLOPE(DZ/DX)

SPANWISE STATION / CHORDWISE STATION	1	2	3	4	5
1	-.07227	-.07472	-.10342	-.11082	-.08727
2	-.07227	-.08667	-.10882	-.10527	-.08727
3	-.07227	-.09772	-.11137	-.09967	-.08727
4	-.07232	-.10507	-.11147	-.09932	-.08727
5	-.07247	-.10937	-.08947	-.09867	-.08727
6	-.07272	-.11107	-.06797	-.09807	-.08727
7	-.07307	-.11067	-.07042	-.09752	-.08727
8	-.07347	-.08622	-.07427	-.09697	-.08727
9	-.07387	-.06237	-.07742	-.09642	-.08727
10	-.07432	-.06497	-.08252	-.09592	-.08727

LOWER SURFACE WING PANEL SLOPE(DZ/DX)

SPANWISE STATION / CHORDWISE STATION	1	2	3	4	5
1	-.10227	-.09982	-.07112	-.06372	-.08727
2	-.10227	-.08787	-.06572	-.06927	-.08727
3	-.10227	-.07682	-.06317	-.07487	-.08727
4	-.10222	-.06947	-.06307	-.07522	-.08727
5	-.10207	-.06517	-.08507	-.07587	-.08727
6	-.10182	-.06347	-.10657	-.07647	-.08727
7	-.10147	-.06387	-.10412	-.07702	-.08727
8	-.10107	-.08832	-.10027	-.07757	-.08727
9	-.10067	-.11217	-.09712	-.07812	-.08727
10	-.10022	-.10957	-.09202	-.07862	-.08727

WING CAMBER SLOPE(DZ/DX)

SPANWISE STATION / CHORDWISE STATION	1	2	3	4	5
1	-.08727	-.08727	-.08727	-.08727	-.08727
2	-.08727	-.08727	-.08727	-.08727	-.08727
3	-.08727	-.08727	-.08727	-.08727	-.08727
4	-.08727	-.08727	-.08727	-.08727	-.08727
5	-.08727	-.08727	-.08727	-.08727	-.08727
6	-.08727	-.08727	-.08727	-.08727	-.08727
7	-.08727	-.08727	-.08727	-.08727	-.08727
8	-.08727	-.08727	-.08727	-.08727	-.08727
9	-.08727	-.08727	-.08727	-.08727	-.08727
10	-.08727	-.08727	-.08727	-.08727	-.08727

WING CHORD LENGTHS(C)

8.22030	6.84619	5.47406	4.10591	2.74785

FORCES AND MOMENTS ON WING-BODY COMBINATION

CD = .03644 CL = .31961 CM = -7.13999

BODY PRESSURE COEFFICIENTS (CP), INCLUDING THE EFFECT OF WING

THETA (DEG) X	0.0000	30.0000	60.0000	90.0000	120.0000	150.0000	180.0000
0.0000	.12638	.13607	.16616	.21578	.27524	.32501	.34454
.7300	.11196	.12129	.15040	.19876	.25703	.30596	.32520
1.4600	.08963	.09797	.12453	.16979	.22539	.27265	.29134
2.1900	.08156	.08911	.11368	.15656	.21022	.25633	.27465
2.9200	.06710	.07400	.09689	.13771	.18955	.23448	.25241
3.6500	.05376	.05992	.08089	.11935	.16914	.21278	.23027
4.3800	.04049	.04587	.06482	.10079	.14842	.19067	.20769
5.1100	.02795	.03250	.04931	.08266	.12800	.16878	.18531
5.8400	.01576	.01944	.03401	.06459	.10749	.14671	.16272
6.5700	.00414	.00691	.01912	.04677	.08710	.12467	.14011
7.3000	-.01976	-.01824	-.00931	.01422	.05081	.08588	.10047
9.2467	-.04166	-.04303	-.04159	-.02744	.00080	.03041	.04313
11.1933	-.04718	-.05153	-.05788	-.05347	-.03395	-.01008	.00068
13.1400	-.02317	-.02981	-.04214	-.04519	-.03229	-.01275	-.00346
15.0867	-.00714	-.01505	-.03072	-.03808	-.02921	-.01244	-.00413
17.0333	.00305	-.03023	-.03075	-.03376	-.01702	.01744	-.00148
18.9800	.00325	-.03347	-.03900	-.02858	.00086	.02940	.00684
20.9267	.00133	-.03776	-.04576	-.02512	.01412	.03968	.01477
22.8733	-.00116	-.04142	-.04992	-.02311	.02218	.04709	.02128
24.8200	-.00299	-.04319	-.05103	-.02211	.02537	.05102	.02575
26.7667	-.00340	-.04239	-.04924	-.02168	.02459	.05142	.02795
28.7133	-.00210	-.03892	-.04509	-.02149	.02103	.04873	.02803
30.6600	.00082	-.03317	-.03942	-.02143	.01570	.04361	.02628
32.6067	.00503	-.02581	-.03302	-.02148	.00944	.03685	.02319
34.5533	.01014	-.01751	-.02651	-.02170	.00280	.02913	.01920
36.5000	.01329	.00562	-.00973	-.01747	-.00993	.00526	.01287

AERODYNAMIC PRESSURE ON THE BODY - - CENTROID OF FINITE ELEMENT AS GENERATED BY NSWC CODE IN THE SAME LOCAL COORDINATE SYSTEM

ELEMENT	X	THETA	PRESSURE	ELEMENT	X	THETA	PRESSURE
90	.844E+01	.280E+03	.428E+00	95	.908E+01	.200E+03	.338E+01
100	.844E+01	.130E+03	.322E+01	105	.908E+01	.498E+02	-.401E-01
110	.104E+02	.340E+03	-.721E+00				

PRESSURE AT CONTROL POINT - UPPER SURFACE

X	Y	PRESSURE
17.5422	2.6691	-.7782E+01
18.3642	2.5691	-.6647E+01
19.1862	2.6691	-.5252E+01
20.0083	2.6691	-.4686E+01
20.8303	2.6691	-.3727E+01
21.6523	2.6691	-.3422E+01
22.4744	2.6691	-.3413E+01
23.2964	2.6691	-.3183E+01
24.1184	2.6691	-.3305E+01
24.9404	2.6691	-.3275E+01
19.8148	4.7292	-.6321E+01
20.4994	4.7292	-.7304E+01
21.1841	4.7292	-.7608E+01
21.8687	4.7292	-.7344E+01
22.5533	4.7292	-.6619E+01
23.2379	4.7292	-.6106E+01
23.9225	4.7292	-.5232E+01
24.6072	4.7292	-.3320E+01
25.2918	4.7292	-.1580E+01
25.9764	4.7292	-.1690E+01
22.0842	6.7864	-.7726E+01
22.6316	6.7864	-.7411E+01
23.1790	6.7864	-.7413E+01
23.7264	6.7864	-.7622E+01
24.2738	6.7864	-.6536E+01
24.8212	6.7864	-.5274E+01
25.3686	6.7864	-.5738E+01
25.9161	6.7864	-.5830E+01
26.4635	6.7864	-.5493E+01
27.0109	6.7864	-.4992E+01
24.3470	8.8377	-.8114E+01
24.7576	8.8377	-.7644E+01
25.1682	8.8377	-.7194E+01
25.5788	8.8377	-.7338E+01
25.9894	8.8377	-.7225E+01
26.4000	8.8377	-.7030E+01
26.8106	8.8377	-.6858E+01
27.2211	8.8377	-.6445E+01
27.6317	8.8377	-.6155E+01
28.0423	8.8377	-.6022E+01
26.5931	10.8738	-.6219E+01
26.8679	10.8738	-.6290E+01
27.1427	10.8738	-.6777E+01
27.4175	10.8738	-.7032E+01
27.6923	10.8738	-.7029E+01
27.9671	10.8738	-.7021E+01

28.2418	10.8738	-.6945E+01
28.5166	10.8738	-.6861E+01
28.7914	10.8738	-.6849E+01
29.0662	10.8738	-.6913E+01

PRESSURE AT CONTROL POINT - LOWER SURFACE

X	Y	PRESSURE
17.5422	2.6691	.8561E+01
18.3642	2.6691	.6694E+01
19.1862	2.6691	.5306E+01
20.0083	2.6691	.4765E+01
20.8303	2.6691	.3756E+01
21.6523	2.6691	.3248E+01
22.4744	2.6691	.3036E+01
23.2964	2.6691	.2454E+01
24.1184	2.6691	.2455E+01
24.9404	2.6691	.2225E+01
19.8148	4.7292	.6861E+01
20.4994	4.7292	.6479E+01
21.1841	4.7292	.6212E+01
21.8687	4.7292	.5262E+01
22.5533	4.7292	.4542E+01
23.2379	4.7292	.3891E+01
23.9225	4.7292	.3273E+01
24.6072	4.7292	.4132E+01
25.2918	4.7292	.5223E+01
25.9764	4.7292	.4650E+01
22.0842	6.7864	.4549E+01
22.6316	6.7864	.4974E+01
23.1790	6.7864	.5161E+01
23.7264	6.7864	.5213E+01
24.2738	6.7864	.6519E+01
24.8212	6.7864	.7748E+01
25.3686	6.7864	.6647E+01
25.9161	6.7864	.5697E+01
26.4635	6.7864	.5096E+01
27.0109	6.7864	.4638E+01
24.3470	8.8377	.3775E+01
24.7576	8.8377	.4270E+01
25.1682	8.8377	.4773E+01
25.5788	8.8377	.4687E+01
25.9894	8.8377	.4878E+01
26.4000	8.8377	.5179E+01
26.8106	8.8377	.5480E+01
27.2211	8.8377	.6040E+01
27.6317	8.8377	.6464E+01
28.0423	8.8377	.6630E+01
26.5931	10.8738	.5447E+01
26.8679	10.8738	.5384E+01
27.1427	10.8738	.4917E+01
27.4175	10.8738	.4686E+01
27.6923	10.8738	.4713E+01
27.9671	10.8738	.4745E+01
28.2418	10.8738	.4849E+01
28.5166	10.8738	.4966E+01
28.7914	10.8738	.5013E+01
29.0662	10.8738	.4989E+01

AERODYNAMIC PRESSURE ON THE WING - - CENTROID OF FINITE ELEMENT AS GENERATED BY NSWC CODE IN THE SAME LOCAL COORDINATE SYSTEM

TOTAL NUMBER OF WINGS 1

WING NO. 1

PRESSURE LOAD ON THE UPPER SURFACE OF WING

ELEMENT	X	Y	PRESSURE	ELEMENT	X	Y	PRESSURE
500	.178E+02	-.267E+01	-.747E+01	501	.200E+02	-.473E+01	-.658E+01
502	.222E+02	-.679E+01	-.762E+01	503	.245E+02	-.884E+01	-.799E+01
504	.267E+02	-.109E+02	-.621E+01	515	.240E+02	-.267E+01	-.328E+01
516	.252E+02	-.473E+01	-.181E+01	517	.264E+02	-.679E+01	-.558E+01
518	.275E+02	-.884E+01	-.620E+01	519	.287E+02	-.109E+02	-.684E+01

PRESSURE LOAD ON THE LOWER SURFACE OF WING

ELEMENT	X	Y	PRESSURE	ELEMENT	X	Y	PRESSURE
525	.178E+02	-.267E+01	.804E+01	526	.200E+02	-.473E+01	.673E+01
527	.222E+02	-.679E+01	.464E+01	528	.245E+02	-.884E+01	.390E+01
529	.267E+02	-.109E+02	.546E+01	540	.240E+02	-.267E+01	.245E+01
541	.252E+02	-.473E+01	.511E+01	542	.264E+02	-.679E+01	.520E+01
543	.275E+02	-.884E+01	.640E+01	544	.287E+02	-.109E+02	.501E+01

.........END OF COMPUTATIONS ....

Section 5
CONCLUSIONS AND RECOMMENDATIONS

The MDAC aerodynamic code has been modified extensively to provide pressure loads on a missile or its components at the centroid of finite elements as specified. The input preparation has been greatly simplified to reduce the effort necessary. In particular, wing parameters such as camber, thickness slope and twist are generated automatically from the wing surface description. Body panel slopes are generated from input by spline interpolation from given profiles. Coordinate systems for the description of finite elements for both body and wing can be different from that used for aerodynamic calculations. Outputs are produced as computer printout and punched cards in the format of NASTRAN bulk data.

The computer program as developed thus provides a unique link between the aerodynamic analysis of a missile at subsonic or supersonic speed and the analysis of its structural response under the aerodynamic load. It reduces considerably the effort required for generating the pressure load at the desired position when used in combination with the NSWC computer code PING or BING. It may be used for analyzing either the component such as wing, body or tail or the complete missile as the situation demands.

The computer code as developed represents a versatile tool for missile analysis. Nevertheless, its range of utilization can be extended with additional effort. It is recommended that the following tasks be considered for future development.

- Interpolation of experimental data for structural analysis: When the measured pressure distributions over a given missile are available from experiment, it is suggested that the present code be used to interpolate the pressure loads and generate the loading at specified centroids of finite elements as input to NASTRAN for analysis. This can be accomplished by incorporating additional

capabilities for inputing test data, performing coordinate transform, interpolating the pressure and generating loads at centroids of elements in punch card form.

- Missile attitude generalization: The present MDAC aero code calculates the aerodynamic loads and missile with a restriction on its attitude. Only the angle of attack given in the pitch plane is considered at the present. For missile attitudes expressed in terms of roll and yaw, modification of the code will be necessary. This addition will in many ways expand the utilization of the code for practical cases of structural analysis of a missile at various attitudes during its flight.
- Aerodynamic loads at high supersonic speed: With the continuing changes in systems requirements as well as improvements in missile technology, the crusing speed of missiles may well be in the high supersonic range of operation. Based on the linear theory the present MDAC aero code is limited to low supersonic speeds ($M_\infty \le 2.5$). For realistic structural analysis of missiles at high supersonic speeds appropriate methods are available for calculating the aerodynamic loads. In view of the current needs of NSWC, it is felt that relatively exact methods (finite difference or method of characteristics) may not be satisfactory in terms of computational effort required. It is suggested that a combination of methods (e.g., second order shock expansion, tangent cone and wedge and Newtonian impact, etc.) be used for analyzing and computing the aerodynamic loads on a missile at high supersonic speed. In References 9 through 12 the above methods are described in some detail and their applications and limitations are discussed.

Section 6

REFERENCES

1. R. H. MacNeal (Editor). The NASTRAN User's Manual, NASA SP-221(01), 1972.

2. H. Ashley and W. P. Rodden. Wing-Body Aerodynamic Interaction. Annual Review of Fluid Mechanics, 1972, Vol 4, P 431-472.

3. F. A. Woodward. Analysis and Design of Wing-Body Combinations at Subsonic and Supersonic Speeds. Journal of Aircraft, 1968, Vol 5, No. 6, P 528-534.

4. P. C. Huang and J. P. Matra, Jr. Planform Input Generating (PING), A NASTRAN Preprocessor for Lifting Surface-Theoretical Development, User's Manual, and Program Listing. NOLTR-73-199, Naval Ordnance Laboratory, White Oak, Silver Spring, Maryland, December 1973.

5. P. C. Huang and J. P. Matra, Jr. Missile Body Input Generator (BING), A NASTRAN Preprocessor; Theoretical Development, User's Manual, and Program Listing. NSWC-NOL-TR-75-9, Naval Surface Weapons Center, White Oak, Silver Spring, Maryland, March 1975.

6. R. L. Harder and R. N. Desmarais. Interpolation Using Surface Splines. Journal of Aircraft, 1972, Vol 9. No. 2, P 189-191.

7. H. W. Carlson. Pressure Distributions at Mach Number 2.05 on a Series of Highly Swept Arrow Wings Employing Various Degrees of Twist and Camber. NASA TN D-1264, Langley Research Center, Langley Station, Hampton, Virginia, May 1962.

8. Gapcynski, John P. and G. J. Landrum. Tabulated Data for a Pressure Distribution Investigation at Mach Number 2.01 of a 45° Sweptback-Wing Airplane Model at Combined Angles of Attack and Sideslip. NASA memo 10-15-58L, November 1958, Langley Research Center, Langley Field, Virginia.

9. Hayes, W. D. and R. F. Probstein. Hypersonic Flow Theory, Vol 1, Inviscid Flows, Second Edition (1966). Academic Press, New York.

10. Savin, R. C. Application of the Generalized Shock-Expansion Method to Include Bodies of Revolution Traveling at High Supersonic Speeds, NACA TN 3345, April 1955.

11. Syvertson, C. A. and D. H. Dennis. A Second-Order Shock-Expansion Method Applicable to Bodies of Revolution Near Zero Lift. NACA Report 1328, 1957.

12. Van Dyke, M. D. Second-Order Slender-Body Theory-Axisymmetric Flow. NASA TR R-47, 1959.

Appendix A
PROGRAM LISTING

PRECEDING PAGE NOT FILMED BLANK

```
CDECK WOODWRD
      OVERLAY(WOOD,0,0)
      PROGRAM WOODWRD(INPUT=200,OUTPUT=200,TAPE5=INPUT,TAPE6=OUTPUT,
     1              TAPE1=200,TAPE2=100,TAPE3=200,TAPE4=100,TAPE8=100,
     2        TAPE9=100,TAPE11=200,TAPE12,PUNCH=200)
C                                                                       WOOD1000
C ********************************************************************* WOOD1010
C *******  SUBSONIC / SUPERSONIC INFLUENCE COEFFICIENT PROGRAM ******** WOOD1020
C ********************************************************************* WOOD1030
C                                                                       WOOD1040
C              AERODYNAMIC COMPUTER CODE FOR ANLAYSIS OF MISSILE WING,
C              BODY, OR COMBINATIONS.  IT IS BASED ON THE WOODWARD
C              PROGRAM DEVELOPED FOR COMPUTING THE PRESSURE LOADS ON
C              MISSILE. IT READS IN THE GEOMETRY OF MISSILE COMPONENTS
C              AND THE FINITE ELEMENT DESCRIPTION FROM NSWC'S PING OR
C              BING AS INPUT, COMPUTES PRESSURE LOAD AND INTERPOLATES,
C              USING SURFACE FIT,FOR PRESSURE ON THE CENTROID OF EACH
C              ELEMENT.  FOR INFORMATION CONTACT  KENNETH K. WANG
C                                                 MCDONNELL-DOUGLAS
C                                                 HUNTINGTON BEACH, CA.
C                                                                       WOOD1090
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1              SYM, KACE, NPOLAR, IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON /INPUT T/ LIN                                              WOOD1120
      COMMON /DSXB/XBS1,XBS2,DSB1,DSB2                                  WOOD1130
      COMMON /TRNSFR/ IOVR10,IOVR60,IOVR70
      COMMON /ENDJOB/ JOBEND
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
C                                                                       WOOD1140
C                                                                       WOOD1250
C     INITIALIZE                                                        WOOD1260
C
      LA=1                                                              WOOD1290
      LB=2                                                              WOOD1300
      LC=3                                                              WOOD1310
      LD=4                                                              WOOD1320
      LIN=5                                                             WOOD1330
      LO=6                                                              WOOD1340
      LF=8                                                              WOOD1350
      LE=9                                                              WOOD1360
      LI=11                                                             WOOD1370
      REWIND1                                                           WOOD1380
      REWIND2                                                           WOOD1390
      REWIND3                                                           WOOD1400
      REWIND4                                                           WOOD1410
      REWIND8                                                           WOOD1420
      REWIND9                                                           WOOD1430
      REWIND11                                                          WOOD1440
C                                                                       WOOD1450
C
C         READ IN CASE SPECIFICATIONS
C
      CALL OVERLAY(4HWANG, 2, 0)
      NPOLAR = 0
C
C         IRW = 0 , DATA ARE NOT SAVED ON TAPE
C         IRW = 1 , DATA ARE SAVED ON TAPE 12 FOR RESTART
C         IRW = 2 , RESTART RUN, BYPASS ALL AERO CALCULATIONS
C
      IF ( IRW  .LT. 2 ) GO TO 100
      CALL OVERLAY(4HWANG, 7, 0)
      GO TO 1100
  100 CONTINUE
C
```

AD-A048 840 MCDONNELL DOUGLAS ASTRONAUTICS CO HUNTINGTON BEACH CALIF F/G 1/1
AERODYNAMIC COMPUTER CODE FOR COMPUTING PRESSURE LOADING ON COM--ETC(U)
JAN 78 K K WANG
UNCLASSIFIED MDC-G7215 NL

2 OF 3
ADA048840

```
      IOVR10 = 0
      CALL OVERLAY(4HWOOD, 1, 0)
C                                                                       WOOD1470
C                                                                       WOOD1690
C     CALL AERODYNAMIC LINK                                             WOOD1700
 1050 CALL AERO                                                         WOOD1710
C                                                                       WOOD1730
C
      WRITE (6,901)
  901 FORMAT (1H0////*..........END OF COMPUTATIONS .......*)
C
C
C             DUMMY READ AND WRITE ON TAPE 9 IN AN UNFORMATED FORM TO
C         MOVE SYSTEM ROUTINE  W,SQ INTO OVERLAY( 0, 0 )  TO AVOID
C         LOADING ERRORS.
C
C         .......  DO NOT REMOVE  ......
C
C
C
      IDUM = 1
      IF ( IDUM  .EQ. 1 ) GO TO 1100
      READ  (9)  DUM
      WRITE (9) DUM
C
 1100 CONTINUE
C
C
C
C ******************                                                    WOOD1760
      STOP                                                              WOOD1770
C ******************                                                    WOOD1780
      END                                                               -
CDECK AERO
      SUBROUTINE AERO                                                   AER01000
C                                                                       AER01010
C ..................................................................... AER01020
C     CONTROL ROUTINE FOR AERODYNAMIC LINKS                             AER01030
C ..................................................................... AER01040
C                                                                       AER01050
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1         NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                       AER01080
      COMMON /TRNSFR/ IOVR10,IOVR60,IOVR70
      COMMON /ENDJOB/ JOBEND
      COMMON /COM60/ ISAVET
C                                                                       AER01150
C                                                                       AER01160
      REWIND NTAPEA                                                     AER01170
      REWIND NTAPEB                                                     AER01180
      REWIND NTAPEC                                                     AER01190
      REWIND NTAPED                                                     AER01200
      REWIND NTAPEE                                                     AER01210
      REWIND NTAPEF                                                     AER01220
C                                                                       AER01230
C                                                                       AER01400
      CALL OVERLAY(4HWOOD,3,0,6HRECALL)
C                                                                       AER01420
      GO TO ( 500, 1070, 2000), KACE
C
C         WING ALONE, KACE = 1
C
  500 CONTINUE
C
      IOVR70 = 1
      CALL OVERLAY(4HWANG, 7, 0, 6HRECALL )
```

```
C
      IOVR70 = 4
      CALL OVERLAY(4HWANG, 7, 0, 6HRECALL )
C
      GO TO 3000
C
C             BODY ALONE, KACE = 2
 1070 CONTINUE
      GO TO 3000
C
C             WING - BODY COMBINATION, KACE = 3
C
 2000 CONTINUE
      IOVR70 = 2
      CALL OVERLAY(4HWANG, 7, 0, 6HRECALL )
      CALL OVERLAY(4HWOOD, 4, 0,6HRECALL)
C
      IOVR70 = 3
      CALL OVERLAY(4HWANG, 7, 0, 6HRECALL )
      CALL OVERLAY(4HWOOD, 5, 0,6HRECALL)
C
      IOVR70 = 4
      CALL OVERLAY(4HWANG, 7, 0, 6HRECALL )
C
 3000 CONTINUE
      IOVR60 = 1
      CALL OVERLAY(4HWOOD, 6, 0,6HRECALL)
C
      IOVR70 = 0
      CALL OVERLAY(4HWANG, 7, 0,6HRECALL)
C
C
C ------------------                                                    AER01760
      RETURN                                                            AER01770
C ------------------                                                    AER01780
C                                                                       AER01790
      END                                                                      -
CDECK FOREOF
      SUBROUTINE FOR EOF (IT)                                           FORE1000
C                                                                       FORE1010
C ..................................................................... FORE1020
C     WRITES E.F. ON A TAPE TO INDICATE THE END OF A DATA SET           FORE1030
C ..................................................................... FORE1040
C                                                                       FORE1050
        INTEGER EOF                                                     FORE1060
C                                                                       FORE1070
      DATA EOF /4HE.F./                                                 FORE1080
C                                                                       FORE1090
      WRITE (IT) EOF                                                    FORE1100
C                                                                       FORE1110
C ------------------                                                    FORE1120
      RETURN                                                            FORE1130
C ------------------                                                    FORE1140
      END                                                                      -
CDECK FSF
      SUBROUTINE FSF (NF, NTAPE, IRR)                                   FSF 1000
C                                                                       FSF 1010
C ..................................................................... FSF 1020
C     READS A TAPE UNTIL IT HAS READ THE CHARACTERS E.F.                FSF 1030
C ..................................................................... FSF 1040
C                                                                       FSF 1050
        INTEGER TW                                                      FSF 1060
        INTEGER EOF                                                     FSF 1070
C                                                                       FSF 1080
      DATA EOF /4HE.F./                                                 FSF 1090
```

```
C
      IRR = 0
      K = 0
      DO 1000  I = 1, 10000
      RE/)(NTAPE) TW
      IF (TW .NE. EOF)  GO TO 1000
      K = K +1
      IF (K .GE. NF)  RETURN
 1000 CONTINUE
      IRR = NF -K
C
C ------------------
      RETURN
C ------------------
      END
CDECK INTURP
      FUNCTION INTURP(IDIC,NDIC,LI,LO)
C
C ...................................................................
C     READ AND WRITE A COMMAND CARD,SEE IF COMMAND (IN THE
C     FIRST WORD) IS IN THE DICTIONARY.
C ...................................................................
C
      INTEGER IDIC(1), COM(20), ICOM
C
      EQUIVALENCE (COM(1),ICOM)
C
C
C     IDIC (INSTEAD OF DIC) IS USED AS AN ARGUMENT
C     SO THAT A FIXED-POINT COMPARISON CAN BE MADE
C
      READ (LI,1040)COM
      IF(LO)1010,1010,1000
 1000 WRITE (LO,1050)COM
 1010 DO 1020 INTERP=1,NDIC
      IF (IDIC(INTERP) .EQ. ICOM)  GO TO 1030
 1020 CONTINUE
C
      INTERP=0
 1030 INTURP=INTERP
C
C ------------------
      RETURN
C ------------------
 1040 FORMAT(20A4)
 1050 FORMAT(/1H0,20A4)
      END
CDECK,CGOP
      OVERLAY(WOOD,1,0)
      PROGRAM CGOP
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1        NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /TRNSFR/ IOVR10,IOVR60,IOVR70
  200 CALL PANEL
C
      END
CDECK PANEL
      SUBROUTINE PANEL
C
C
      COMMON/ MAIN / LA,NTAPEB,NTAPEC,NTAPED,LE,LF,LI,LO,NBODY,NWING,
     1        XMACH, SYM, KACE, NPOLAR, IRW
      COMMON /XYWING/ XW(13,11),YW(11),ZW(11)
      COMMON/DSXB / XBS1, XBS2, DSB1, DSB2
      DIMENSION XB(51),RB(51),ZB(51),THETA(11),X(51),Y(11),Z(11),BC(10),
```

```
     1XBAR(50),YBAR(10),ZBAR(10),XC(50),THET(11),CHORD(50),AREA(50,10),
     2KW(10),ZBR(110,10),XBR(110,10),YBR(110,10),CW(110,10),THETW(10)
     3,ARN(110,10),XCW(110,10)
C
      REWIND NTAPEC
      XCPT = 0.95
      I=1
      ZO=0.
      ON=1.
      IF ( NBODY ) 100, 100, 10
   10 READ (LI,1010) BODYS, BTHET, PLANE, RP
      NBODYS=BODYS
      NTHET=BTHET
      NPLANE=PLANE
      NXLE=0
      READ  (LI,1010) ( XB(N), N=1,NBODYS)
      READ  (LI,1010) ( RB(N), N=1,NBODYS)
      READ  (LI,1010) ( ZD(N), N=1,NBODYS)
      READ  (LI,1010) ( THETA(N), N=1,NTHET )
      READ  (LI,1010) XBS1, XBS2, DSB1, DSB2
      DO 15 N=1,NTHET
   15 THETA(N)=THETA(N)/57.2957795
      IF(PLANE) 300,300,20
C
C         BODY-WING CASE, REQUIRES PANELED BODY
C
   20 CONTINUE
      READ  (LI,1010) ( X(N), N=1,NPLANE)
      ALPHAA=0
      NPLN1=NPLANE-1
      NROW=NTHET-1
      IF(RP) 22,22,28
C
C         NON-CIRCULAR BODY
C
   22 CONTINUE
      READ  (LI,1010) ( Y(N), N=1,NTHET)
      READ  (LI,1010) ( Z(N), N=1,NTHET)
      GO TO 35
C
C         CURCULAR BODY
C
   28 DO 30 N=1,NTHET
      Y(N)=RP*SIN(THETA(N))
   30 Z(N)=RP*COS(THETA(N))
   35 DO 40 N=1,NROW
      YBAR(N)=(Y(N)+Y(N+1))/2.
      ZBAR(N)=(Z(N)+Z(N+1))/2.
   40 BC(N)=SQRT((Y(N+1)-Y(N))**2+(Z(N+1)-Z(N))**2)
      DO 50 M=1,NPLN1
      XBAR(M)=(X(M+1)+X(M))/2.
      CHORD(M)=X(M+1)-X(M)
   50 XC(M)=X(M)+XCPT*CHORD(M)
      DO 60 N=1,NROW
   60 THET(N)=ATAN2((Z(N+1)-Z(N)),(Y(N+1)-Y(N)))
      WRITE(LO,900)
      WRITE(LO,910)
      NP=0
      DO 70 N=1,NROW
      DO 70 M=1,NPLN1
      NP=NP+1
      WRITE(LO,920)  NP,I,X(M),Y(N),Z(N),X(M),Y(N+1),Z(N+1),
     1X(M+1),Y(N),Z(N),X(M+1),Y(N+1),Z(N+1)
   70 WRITE (NTAPEC) NP,I,X(M),Y(N),Z(N),X(M),Y(N+1),Z(N+1),
     1X(M+1),Y(N),Z(N),X(M+1),Y(N+1),Z(N+1)
```

```
      WRITE(LO,940)
      WRITE(LO,950)
      NP=0
      DO 80 N=1,NROW
      DO 80 M=1,NPLN1
      NP=NP+1
      AREA(M,N)=CHORD(M)*BC(N)
      WRITE(LO,960)  NP,XBAR(M),YBAR(N),ZBAR(N),XC(M),YBAR(N),ZBAR(N),
     1AREA(M,N),THET(N  ),ZO,CHORD(M)
   80 WRITE (NTAPEC) NP,XBAR(M),YBAR(N),ZBAR(N),XC(M),YBAR(N),ZBAR(N),
     1AREA(M,N),THET(N  ),ZO,CHORD(M)
      WRITE (NTAPEC) NPLN1,XCPT
      NBODY=NP
      IF(NWING)500,500,120
  100 IF(NWING)500,500,110
C
C        BODY ONLY OR BODY-WING CASE
C
  110 CONTINUE
      NBODY=0
      PLANE=0.
  120 NP=0
      WRITE(LO,970)
      WRITE(LO,980)
      DO 200 L=1,NWING
      CALL WINGP(NB,NC,YO,ZA,DIH)
      THETW(L)=DIH
      WRITE(LO,1020) L,YO,ZA,DIH
      NB1=NB-1
      NC1=NC-1
      KW(L)=NB1*NC1
      DO 150 N=1,NB1
      DO 150 M=1,NC1
      NP=NP+1
      WRITE (NTAPEC) NP,I,XW(M,N),YW(N),ZW(N),XW(M,N+1),YW(N+1),ZW(N+1),
     1XW(M+1,N),YW(N),ZW(N),XW(M+1,N+1),YW(N+1),ZW(N+1)
      WRITE(LO,920)  NP,I,XW(M,N),YW(N),ZW(N),XW(M,N+1),YW(N+1),ZW(N+1),
     1XW(M+1,N),YW(N),ZW(N),XW(M+1,N+1),YW(N+1),ZW(N+1)
      K=M+(N-1)*NC1
      CR=XW(M+1,N)-XW(M,N)
      CT=XW(M+1,N+1)-XW(M,N+1)
      RY=(1.+CT/(CR+CT))/3.0
      RY1=1.0-RY
      XWL=RY*XW(M,N+1)+RY1*XW(M,N)
      XWT=RY*XW(M+1,N+1)+RY1*XW(M+1,N)
      CW(K,L)=XWT-XWL
      YDUM=SQRT((YW(N+1)-YW(N))**2+(ZW(N+1)-ZW(N))**2)
      ARW(K,L)=YDUM*.5*(CR+CT)
      XBR(K,L)=(XWL+XWT)/2.
      YBR(K,L)=YW(N+1)*RY+YW(N)*RY1
      ZBR(K,L)=ZW(N+1)*RY+ZW(N)*RY1
  150 XCW(K,L)=XCPT*XWT+(1.0-XCPT)*XWL
  200 CONTINUE
      WRITE(LO,990)
      WRITE(LO,1000)
      NP=0
      DO 250 L=1,NWING
      KL=KW(L)
      DO 250 K=1,KL
      NP=NP+1
      WRITE(LO,960) NP,XBR(K,L),YBR(K,L),ZBR(K,L),XCW(K,L),YBR(K,L),
     1ZBR(K,L),ARW(K,L),THETW(L),ZO,CW(K,L)
  250 WRITE(NTAPEC) NP,XBR(K,L),YBR(K,L),ZBR(K,L),XCW(K,L),YBR(K,L),
     1ZBR(K,L),ARW(K,L),THETW(L),ZO,CW(K,L)
      NWING=NP
```

```
      WRITE (NTAPEC) NC1,XCPT
      IF  (PLANE.GT.0.) WRITE(NTAPEC) NTHET
      IF  (PLANE.GT.0.) WRITE(NTAPEC) (THETA(N),N=1,NTHET)
      IF(PLANE) 500,500,400
  300 NBODY=-1
      REWIND NTAPED
      WRITE(NTAPED) NDUM,NDUM,NDUM,NTHET
      WRITE(NTAPED) (THETA(N),N=1,NTHET)
      REWIND NTAPED
  400 REWIND NTAPEB
      WRITE (NTAPEB) ON,ON,ZO,ZA,DSB1,DSB2
      WRITE (NTAPEB) NBODYS,NXLE
      WRITE (NTAPEB) (XB(N),RB(N),ZD(N),N=1,NBODYS)
      REWIND NTAPEB
  500 REWIND NTAPEC
      NPANEL=NWING+NBODY
      RETURN
  900 FORMAT (1H1,9X,*BODY PANEL CORNER POINT LOCATIONS EXPRESSED IN *,
     A          *THE AERO COORDINATE SYSTEM*/10X,
     186H1 AND 2 INDICATE BODY PANEL LEADING-EDGE POINTS, 3 AND 4 INDICA
     2TE TRAILING-EDGE POINTS)
  910 FORMAT (1H0,5X,5HPANEL,    4(8X,1HX,8X,1HY,8X,1HZ)/
     15X,2HNO,2X,5HPARTS,6X,
     2    3(1H1,8X),3(1H2,8X),3(1H3,8X),3(1H4,8X)//)
  920 FORMAT (1H ,3X,I3,3X,I1,3X,12F9.5)
  930 FORMAT (1H ,13X,12F9.5)
  940 FORMAT (1H1,9X,*BODY PANEL CENTROID AND CONTROL POINT LOCATIONS*,
     1          * EXPRESSED IN THE AERO COORDINATE SYSTEM*)
  950 FORMAT (1H0,5HPANEL,6X,2(1HX,8X,1HY,8X,1HZ,8X),5X,4HAREA,9X,
     16HTHETA-,6X,6HALPHA-, 5X,5HCHORD/13X,3(1HC,8X),3(2HCP,7X),
     217X,6HINCLIN,6X,5HINCID//)
  960 FORMAT (1H ,1X,I3,4X,6F9.5,5X,F9.5,5X,F12.5,2F9.5)
  970 FORMAT (1H1,9X,*WING PANEL CORNER POINT LOCATIONS EXPRESSED IN *,
     A          *THE AERO COORDINATE SYSTEM*/10X,
     186H1 AND 2 INDICATE WING PANEL LEADING-EDGE POINTS, 3 AND 4 INDICA
     2TE TRAILING-EDGE POINTS)
  980 FORMAT (1H0,5X,5HPANEL,    4(8X,1HX,8X,1HY,8X,1HZ)/
     15X,2HNO,2X,5HPARTS,6X,
     2    3(1H1,8X),3(1H2,8X),3(1H3,8X),3(1H4,8X)//)
  990 FORMAT (1H1,9X,*WING PANEL CENTROID AND CONTROL POINT LOCATIONS *,
     A          *EXPRESSED IN THE AERO COORDINATE SYSTEM*)
 1000 FORMAT (1H0,1X,5HPANEL,6X,2(1HX,8X,1HY,8X,1HZ,8X),5X,4HAREA,12X,
     16HTHETA-,3X,6HALPHA-, 5X,5HCHORD/14X,3(1HC,8X),3(2HCP,7X),
     220X,5HDIHED,4X,6HCAMBER,4X//)
 1010 FORMAT(7F10.0)
 1020 FORMAT(1H0,4X,9HWING NO.  ,I2,5X,9H YPIVOT =,F10.3,5X,9H ZPIVOT =,
     1F10.3,5X,16H WING DIHEDRAL =,F10.3,1X,7HRADIANS,1X/)
      END
CDECK WINGP
      SUBROUTINE WINGP(NB,NC,YO,ZA,DIH)
      COMMON/ MAIN / LA,NTAPEB,NTAPEC,NTAPED,LE,LF,LI,LO,NBODY,NWING,
     1          XMACH, SYM, KACE, NPOLAR, IRW
      COMMON /XYWING/ XY(13,11),YW(11),ZW(11)
      DIMENSION XW(12),DY(11)
      READ (LI,1010) XO,YO,CR,A,B,C
      READ (LI,1010) XNB,XNC,CASE,ZA
      NB=XNB
      NC=XNC
      NCA=CASE
      PERCT = 0.1
C
      CRR = CR - XO
      DCR = ( 1.0 - PERCT ) * CRR / ( XNC - 2.0 )
      XW(1)=XO
      XW(2) = XW(1) + PERCT* CRR
```

```
      DO 145 M=3,NC
      XW(M) = XW(M-1) + DCR
  145 CONTINUE
      BY = B - YO
      DBY=BY/(XNB-1.)
      YW(1)=YO
      DO 165 N=2,NB
      YW(N) = YW(N-1) + DBY
  165 CONTINUE
      SLE=(A-XO)/(B-YO)
      STE=(C-CR)/(B-YO)
      IF(SLE-STE)210,230,210
  210 CONTINUE
      DX = ( C-A ) / ( CR - XO )
      DO 215 M=1,NC
      XY(M,NB) = A + DX*( XW(M) - XO )
  215 CONTINUE
      DO 220 N=1,NB
      DY(N)=(YW(N)-YO)/(B-YO)
      DO 220 M=1,NC
      XY(M,N) = XW(M) + ( XY(M,NB) - XW(M) ) * DY(N)
  220 CONTINUE
      GO TO 250
  230 DO 240 N=1,NB
      DY(N)=YW(N)-YO
      DO 240 M=1,NC
      XY(M,N) = XW(M) + SLE*DY(N)
  240 CONTINUE
  250 CONTINUE
      CALL WROTAT(NB,YO,ZA,DIH)
      RETURN
 1010 FORMAT(7F10.0)
      END
CDECK WROTAT
      SUBROUTINE WROTAT(NB,YO,ZA,WNGDIH)
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1          SYM, KACE, NPOLAR, IRW
      COMMON /XYWING/ XY(13,11),YW(11),ZW(11)
      DIMENSION DYPIVT(11)
      WNGDIH=0.
      READ (11,1020)  PIVOT, WNGDIH
      YPIVOT=YO
      ZPIVOT=ZA
C  PIVOT, DIHEDRAL INDICATOR
C  YPIVOT, ZPIVOT, PIVOT POINT
C  WNGDIH, DIHEDRAL ANGLE
      DO 50 I=1,NB
   50 ZW(I)=ZA
      IF(PIVOT.EQ.0.0)RETURN
      DO 100 I=1,NB
      DYPIVT(I)=YW(I)-YPIVOT
 100  CONTINUE
      WNGDIH=WNGDIH/57.2957795
      COSDIH=COS(WNGDIH)
      SINDIH=SIN(WNGDIH)
      DO 200 I=1,NB
      YW(I)=YPIVOT+COSDIH*DYPIVT(I)
      ZW(I)=ZPIVOT+SINDIH*DYPIVT(I)
 200  CONTINUE
 1020 FORMAT (7F10.4)
      RETURN
      END
CDECK,PRELIM
      OVERLAY(WANG, 2, 0)
      PROGRAM PRELIM
```


MCDONNELL DOUGLAS

```
C
C
C           READ INPUT,  WRITE  ON TAPE 11 FOR SUBSEQUENT COMPUTATION
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING,XMACHN,
     1        SYMM, KACE, NPOLAR, IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON/DEFAU / BODY, BODYS, BTHET, CPCALC,WCASE, NRUN, DADEG, ARB,
     1           PLANE, POLAR, RP, XNACEL, SYM, THICK,  WING, XMACH,
     2           XNBB,XNCC,PINF,IPUNCH,SID(20), CKASE, ARW
      COMMON/ BODY2 / NBPFL, NBODYS, NTHET, XB(51), RB(51), ZD(51),
     1           THETA(11), RPP, XBPANL(20)
      COMMON/ NACEL /NACELP, XN(51,2), RN(51,2), ZN(51,2), ARN(2)
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/INPUT / TX(4, 100)
      COMMON/ POINTP/ TY(4), TYP(4), TYPP(4)
      COMMON/ POINTS / NPTS, NVAR
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1           NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2             , PIVOT(10), DIHED(10)
      INTEGER TITLE(20)
      DIMENSION NBC(2, 6)
      DATA NBC / 4, 9, 4, 6, 3, 7, 3, 6, 3, 4, 2, 5 /
C
      NAMELIST/ UNIFID /ICASE,NTRANS,IPUNCH,   IRW, POLAR,ISOLID,   SID,
     1             NBPFL,    XB,    RB,    ZD,    RP,  NWPI,
     2             NWING, IFORM, PIVOT, THICK, DIHED,   XWI,   YWI,   ZWI,
     3            XNACEL,NACELP,    XN,    RN,    ZN,
     4             XMACH,  PINF, DADEG,   ARB,   ARW,   ARN
C
C
C           READ IN DESCRIPTIVE TITLE
C
      READ (5,1000)  TITLE
      WRITE (6,1010) TITLE
C
C           SET DEFAULT VALUES
C
      CALL DEFAUT
C
C           READ NAMELIST INPUT FOR UNIFIED COMPUTER CODE
C
      READ ( 5, UNIFID )
      WRITE ( 6, UNIFID )
C
      NSTR = NWING - 2
      IF ( NWING  .LE. 2 ) GO TO 30
      XNBB = FLOAT( NBC( 1, NSTR ) )
      XNCC = FLOAT( NBC(2,NSTR) )
   30 CONTINUE
      XNB = XNBB
      XNC = XNCC
      NB = XNB
      NC = XNC
      NBODY = 0
      NCHORD = NC
      NSPAN = NB
C
C        FIRST RUN IRW .LT. 2
C        RESTART RUN IRW = 2
C
      IF ( IRW  .EQ. 2 ) GO TO 500
C
C        READ IN COORDINATE SYSTEM SPECIFICATION FOR BODY AND WING AND
C        COMPUTE TRANSFORMATION MATRICES.
```

```
C
      IF ( ICASE  .NE. 1 ) NBODY = 1
      ANACEL = ABS( XNACEL )
      NACELS= IFIX(ANACEL)
      NW2 = 0
      IF ( ICASE  .EQ. 2 ) GO TO 45
      DO 40 I=1,NWING
      NCORD = 1
      IF ( IFORM(I) .EQ. 1 ) NCORD = 2
      NW2 = NW2 + NCORD
   40 CONTINUE
   45 CONTINUE
      NS = NBODY + NW2 + NACELS
      DO 50 I=1,NS
      NSYM = I
      CALL CORDTR
   50 CONTINUE
C
      IF ( ICASE  .EQ. 2 ) GO TO 70
C
C        TRANSFORM WING COORDINATES TO THAT OF AERO
C
      NSYM = NBODY
      DO 60 I = 1,NWING
      NSYM = NSYM + 1
      NW = 2 * I - 1
      CALL WTRANS( NW, I )
      CALL CORTRN( NW, I )
      IF ( IFORM(I)  .NE. 1 ) GO TO 60
      NSYM = NSYM + 1
      NW = NW + 1
      CALL WTRANS( NW, I )
      CALL CORTRN( NW, I )
C
   60 CONTINUE
      IF ( NACELS  .EQ. 0 ) GO TO 70
C
C        TRANSFORM THE NACEL AXIS XN TO THAT OF THE BODY AXIS
C
      DO 51 K = 1, NACELS
      NSYM = 1 + NW2 + K
      NT = NTRANS( NSYM )
      DO 51 ITRAN = 1, NT
      DO 51 J = 1, NACELP
      XN(J) = XN(J) + VA( 1, ITRAN, NSYM )
   51 CONTINUE
C
   70 CONTINUE
C
C        PREPARE INPUTS AND WRITE ON TAPE 11 FOR AERO CODE
C
C        ICASE = 1 ,  WING  ONLY
C        ICASE = 2 ,  BODY  ONLY
C        ICASE = 3 ,  COMPLETE MISSILE
C
      GO TO ( 100, 200, 300 ) , ICASE
C
C        WING ONLY CASE
C
  100 CONTINUE
      KACE = 1
      CALL WINGIN
      GO TO 500
C
  200 CONTINUE
```

```
C
C        BODY ONLY CASE
C
      KACE = 2
      NBODY = 1
      XNACEL = 0.0
      NWING = 0.0
      PLANE = 0.0
      CALL BODYIN
      GO TO 500
C
  300 CONTINUE
C        COMPLETE MISSILE CASE
C
      KACE = 3
      CALL BANWIN
C
  500 CONTINUE
C
C
 1000 FORMAT(20A4)
 1010 FORMAT(1H1,20A4)
 1020 FORMAT (7F10.0)
C
C
      END
CDECK,BODYPN
      SUBROUTINE BODYPN
C
C
C           PANEL BODY SECTION TO CORRESPONDING WITH THE WING PANEL FOR
C           AERODYNAMIC COMPUTATIONS
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1         SYM, KACE, NPOLAR, IRW
      COMMON/ BODY2 / NBPFL, NBODYS, NTHET, XB(51), RB(51), ZD(51),
     1         THETA(11), RP, XN(20)
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/DEFAU / BODY, BODYS, BTHET, CPCALC, WCASE, NRUN, DADEG, ARB
     1         ,PLANE, POLAR, RPP,XNACEL, SYMM, THICK, WING, XMACHH, XNB
     2         , XNC, PINF, IPUNCH, CKASE, ARW
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1         NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2         , PIVOT(10), DIH(10)
C
      PERCT = 0.1
      NC = XNC
C
C
C           DETERMINE BODY PANEL RADIUS  RP
C
      YMIN = YWI( 1,1)
      NW2 = 2 * NWING
      XMIN = XWI( 1, 1)
      XMAX = XWI( 2,1)
      IF ( NWING  .EQ. 1 ) GO TO 305
      DO 200 I=1,NW2,2
      YMIN = AMIN1( YWI(1,I), YMIN )
  200 CONTINUE
C
      DO 300 I=1,NW2,2
      XMIN = AMIN1( XWI(1,I), XMIN )
      XMAX = AMAX1( XWI(2,I), XMAX )
  300 CONTINUE
C
```

```
  305 CONTINUE
      RP = YMIN
      RPP = YMIN
C
      DO 310 I=1,NW2,2
      KMAX = I
      IF ( XMAX - XWI(2,I)  .LT. 1.0E-3 ) GO TO 315
  310 CONTINUE
  315 CONTINUE
      DO 320 I=1,NW2,2
      KMIN = I
      IF ( XMIN - XWI(1,I)  .LT. 1.0E-3 ) GO TO 325
  320 CONTINUE
  325 CONTINUE
C
C
C          COMPUTE THE BODY PANEL STREAMWISE COORDINATES
C
      NC3 = ( NC-1) / 2 + 2
      IF ( KMAX  .EQ. KMIN ) NC3 = NC
      NF = 2
      IF ( KMAX  .EQ.  KMIN ) NF = 1
      XNN = XWI( 2,KMIN) - XWI( 1,KMIN )
      DXN = XNN/ ( XNC - 1.0 )
      XC = ( 1.0 - PERCT ) * XNN/ ( NC - 2 )
C
      XN(1) = XWI( 1, KMIN) - DXN
      XN(2) = XWI( 1, KMIN)
      XN(3) = XN(2) + XC + PERCT * XNN
      DO 420 I=4,NC3
      XN(I) = XN(I-1) + X' * NF
  420 CONTINUE
C
      IF ( KMAX  .EQ.  KMIN ) GO TO 500
C
      NC4 = NC3 + 3
      NC5 = NC4 + 1
      NC6 = NC5 + 1
      NC7 = NC3 * 2
      XM = XWI( 2,KMAX) - XWI( 1, KMAX)
      DXM = XM / ( XNC - 1.0 )
      XC = ( 1.0 - PERCT ) * XM / ( NC-2 )
C
      XN( NC3+1) = ( XWI( 1,KMAX)- XWI(2,KMIN) )/ 3.0 + XWI(2,KMIN)
      XN(NC3+2) = ( XWI(1,KMAX) - XWI(2,KMIN) ) / 3.0 + XN(NC3+1)
      XN( NC4 ) = XWI( 1 , KMAX )
      XN(NC5) =XN(NC4) + XC + PERCT*XM
      DO 440 I= NC6, NC7
      XN(I) = XN(I-1) + NF * XC
  440 CONTINUE
      XN(NPLANE) = XB(NBODYS)
      GO TO 700
C
  500 CONTINUE
      DXR = ( XB(NBODYS) - XN(NC3) ) / ( NPLANE - NC3 )
      NC4 = NC3 + 1
      DO 600 I= NC4, NPLANE
      XN(I) = XN(I-1) + DXR
  600 CONTINUE
  700 CONTINUE
C
      RETURN
      END
CDECK,CORTRN
      SUBROUTINE CORTRN( NW, IW )
```

```
C
C              TRANSFORM COORDINATES FROM THE LOCAL SYSTEM TO THE AERO
C          SYSTEM  USING THE TRANSFORM MATRIX BTFM
C
C
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1          NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2          , PIVOT(10), DIH(10)
      DIMENSION TX(3), X(3)
C
      NWP = NWPI( NW )
      IF ( IFORM( IW )  .EQ. 2 ) NWP = 4
      NT = NTRANS( NSYM )
      DO 250 ITRAN = 1, NT
C
      DO 220 L = 1, NWP
      X(1) = XWI( L, NW )
      X(2) = YWI( L, NW )
      X(3) = ZWI( L, NW )
C
      DO 100 J=1,3
      SUM = 0.0
      DO 120 K=1,3
      JK = K + ( J-1) * 3
      SUM = SUM + X(K) * BTFM( JK, ITRAN, NSYM )
  120 CONTINUE
      TX(J) = SUM
  100 CONTINUE
C
      XWI(L,NW) = TX(1) + VA(1, ITRAN, NSYM )
      YWI(L,NW) = TX(2) + VA(2, ITRAN, NSYM )
      ZWI(L,NW) = TX(3) + VA(3, ITRAN, NSYM )
  220 CONTINUE
C
  250 CONTINUE
C
      RETURN
      END
CDECK,BODYPF
      SUBROUTINE BODYPF( XB, RB, ZD, THETA, NP, NTHET, NBODYS, XBS1,
     1          XBS2, DSB1, DSB2 )
C
C         GENERATES BODY PROFILE R = R(X), ZD = ZD(X) AND MERIDIAN ANGLE
C               THETA.
C         COMPUTES SLOPES DSB1, DSB2
C
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING,XMACHN,
     1          SYMM, KACE, NPOLAR, IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON/INPUT / TX(4, 100)
      COMMON/ POINTP/ TY(4), TYP(4), TYPP(4)
      COMMON/ POINTS/ NPTS, NVAR
      DIMENSION XB(51), RB( 51), ZD( 51), THETA( 11)
C
      NPTS = NP
C
C         SPLINE FIT INPUT PROFILE AFTER COORDINATE TRANSFORM.
C
      ICAMBR = 0
      DO 100 I=1,NPTS
      TX(1,I) = XB(I)
      TX(2,I) = RB(I)
      IF ( ZD(I)  .LT. 1.E-10 ) GO TO 100
```

```
      ICAMBR = 1
      TX(3,I) = ZD(I)
  100 CONTINUE
C
      NVAR = 3
      IF ( ICAMBR  .EQ. 0 ) NVAR = 2
      CALL SPLFIT
C
C        FOR IMPROVED ACCURACY, FIRST 20 PERCENT OF BODY IS DIVIDED INTO
C        10 EQUAL SECTIONS, THE REMAINING PART IS DIVIDED INTO 15 EQUAL
C        SECTIONS.
C
      N1 = 11
      XLAST = XB(NPTS)
      BODY1 = NBODYS - 1
      DX = XLAST * 0.2 / 10.0
      DO 200 I=1,N1
      AI = I - 1
      TY(1) = DX * AI
      XB(I) = TY(1)
      CALL POINT(1)
      RB(I) = TY(2)
      IF ( ICAMBR  .EQ. 0 ) GO TO 200
      ZD(I) = TY(3)
  200 CONTINUE
C
      XS1 = N1 - 1
      N2 = N1 + 1
      DX = ( XLAST - XB(N1) ) / ( BODY1 - XS1 )
      DO 250 I=N2, NBODYS
      XB(I) = XB(I-1) + DX
      TY(1) = XB(I)
      CALL POINT(1)
      RB(I) = TY(2)
      IF( ICAMBR  .EQ. 0 ) GO TO 250
      ZD(I) = TY(3)
  250 CONTINUE
C
C
C        CALCULATE THE SLOPE AND CAMBER OF BODY AT POINT MIDWAY BETWEEN
C              STATION.
C
      DO 275 I=2,NBODYS
      TY(1) = ( XB(I-1) + XB(I) ) * 0.5
      CALL POINT(1)
      DRDX(I-1) = TYP(2) / TYP(1)
      IF ( ICAMBR  .EQ. 0 ) GO TO 275
      DZDX(I-1) = TYP(3) / TYP(1)
  275 CONTINUE
      NBS1 = NBODYS - 1
      DRDX(NBODYS) = (RB(NBODYS) - RB(NBS1) )/ ( XB(NBODYS) - XB(NBS1) )
C
      TY(1) = XB(1)
      CALL POINT(1)
      DSB1 = TYP(2) / TYP(1)
      TY(1) = XB(51)
      CALL POINT(1)
      DSB2 = TYP(2) / TYP(1)
C
C        GENERATE THETA
      DTH = 180.0 / FLOAT( NTHET - 1 )
      DO 300 I=1,NTHET
      AI = I - 1
      THETA(I) = DTH * AI
  300 CONTINUE
```

```
      XBS1 = 1.0
      XBS2 = NBODYS
C
      RETURN
      END
CDECK,DEFAUT
      SUBROUTINE DEFAUT
C
      COMMON/DEFAU / BODY, BODYS, BTHET, CPCALC,WCASE, NRUN, DADEG, ARB,
     1         PLANE, POLAR, RP, XNACEL, SYM, THICK,  WING, XMACH,
     2         XNB, XNC, PINF, IPUNCH, SID(20), CKASE, ARW
C
      ARW = 0.0
      BODY = 0.0
      BODYS = 26.0
      BTHET = 7.0
      CPCALC = 0.0
      DADEG = 0.0
      DIH = 0.0
      NRUN = 1.
      IPUNCH = 0
      CKASE = 2
      PINF = 14.7
      PIVOT = 0.0
      PLANE = 15.0
      POLAR = 0.0
      RFAREA = 0.0
      RP = 0.0
      DO 100 I=1,20
      SID(I) = 1.0
  100 CONTINUE
      SYM = 1.0
      THICK = 1.0
      THKOMP = 1.0
      TWIST = 0.0
      WING = 0.0
      WCASE = 3
      XMACH = 2.0
      XNACEL = 0.0
      XNB = 6.0
      XNC = 11.0
      XP = 0.0
      ZA = 0.0
      ZP = 0.0
C
      RETURN
      END
CDECK CORDTR
      SUBROUTINE CORDTR
C
C
C              READ IN COORDINATE SPECIFICATIONS VECTORS VA, VB, AND VC
C          AND COMPUTES THE TRANSFORMATION MATRICES  BTFM  FOR EACH
C          COORDINATE SYSTEM
C
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
C
      DIMENSION ICOORD(3), VB( 3, 5), VC( 3, 5)
C
      DATA ICOORD/1HR, 1HC, 1HS /
C
C
C              COMPUTE COORDINATE SYSTEM TRANSFORM MATRICES
C
      NTN = NTRANS(NSYM)
```

```
      DO 100 ITRAN = 1, NTN
C
      READ (5,901) ID, ICHECK,NM, (VA(I,ITRAN,NSYM), I=1,3),(VB(I,ITRAN)
     1        , I=1,3)
      READ (5,902) (VC( I, ITRAN), I=1,3)
      IF ( ITRAN  .GT. 1 ) GO TO 50
      IF ( ICOORD(1) .EQ. ICHECK ) ISYM( NSYM ) = 0
      IF ( ICOORD(2) .EQ. ICHECK ) ISYM( NSYM ) = 1
      IF ( ICOORD(3) .EQ. ICHECK ) ISYM( NSYM ) = 2
C
   50 CONTINUE
      CALL TRANSM( VA(1,ITRAN,NSYM),VB(1,ITRAN),VC(1,ITRAN),BTFM(1,
     1          ITRAN, NSYM)  )
  100 CONTINUE
C
C
  901 FORMAT ( A5, A1, 2X, I8, 8X, 6F8.2 )
  902 FORMAT ( 8X, 3F8.2)
C
      RETURN
      END
CDECK,BODYIN
      SUBROUTINE BODYIN
C
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING,XMACHN,
     1        SYMM, KACE, NPOLAR, IRW
      COMMON/ BODY2 / NBPFL, NBODYS, NTHET, XB(51), RB(51), ZD(51),
     1          THETA(11), RPP, XBPANL(20)
      COMMON/BODYSP/ DRDX(51), DZDX(51)
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/DEFAU / BODY, BODYS, BTHET, CPCALC,WCASE, NRUN, DADEG, ARB,
     1          PLANE, POLAR, RP, XNACEL, SYM, THICK,  WING, XMACH,
     2          XNB, XNC, PINF, IPUNCH, SID(20), CKASE, ARW
      COMMON/WINGFM/XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),
     1       NCHORD,NSPAN,ISOLID,NPLANE,ICAMBR,ITWIST,ITHICK,
     2       IFORM(10), PIVOT(10), DIH(10)
C
      NBODY = 1
      NBODYS = IFIX( BODYS )
      NSYM = 1
      NTHET = IFIX( BTHET )
      NWING = 0
      PLANE = 0.0
      THKOMP = 1.0
      WING = 0.0
      SOLID = 0.0
      THICK = 0.0
      PUNCH = FLOAT( IPUNCH )
C
C
C          READ IN BODY PROFILE AND COMPUTE BODY RADIUS DISTRIBUTION AT
C          OPTIMUM LOCATIONS.
C
      CALL BODYPF( XB, RB, ZD, THETA, NBPFL, NTHET, NBODYS, XBS1, XBS2,
     1          DSB1, DSB2 )
C
      BODYS = FLOAT( NBODYS )
C
C          WRITE ON TAPE 11 FOR AERODYNAMIC ANALYSIS.
C
      REWIND  11
      WRITE (11,1020) BODYS, BTHET, PLANE, RP
      WRITE ( 11,1020) ( XB(N), N=1,NBODYS)
      WRITE ( 11,1020) ( RB(N), N=1,NBODYS)
```

```
      WRITE ( 11,1020) ( ZD(N), N=1,NBODYS)
      WRITE ( 11,1020) ( THETA(N), N=1,NTHET)
      WRITE (11,1020) XBS1,XBS2,DSB1,DSB2
      WRITE (11,1020) XMACH, SYM, THKOMP
      WRITE (11,1020) CKASE,CPCALC,POLAR,THICK,XNACEL,PINF,PUNCH,SOLID,
     1          WING, (SID(K), K=1,20), (IFORM(K), K=1,10)
      WRITE ( 11,1020) ARB, DADEG
      REWIND 11
C
 1020 FORMAT (7F10.4)
C
      RETURN
      END
CDECK,BANWIN
      SUBROUTINE BANWIN
C
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING,XMACHN,
     1         SYMM, KACE, NPOLAR, IRW
      COMMON/ BODY2 / NBPFL, NBODYS, NTHET, XB(51), RB(51), ZD(51),
     1         THETA(11), RPP, XBPANL(20)
      COMMON/BODYSP/ DRDX(51), DZDX(51)
      COMMON/ NACEL /NACELP, XN(51,2), RN(51,2), ZN(51,2), ARN(2)
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/DEFAU / BODY, BODYS, BTHET, CPCALC,WCASE, NRUN, DADEG, ARB,
     1         PLANE, POLAR, RP, XNACEL, SYM, THICK,  WING, XMACH,
     2         XNB, XNC, PINF, IPUNCH, SID(20), CKASE, ARW
      COMMON/WANGLE/ ALPHAC(55,10), ALPHAT(55,10), ARWT(55,10)
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1         NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2           PIVOT(10), DIH(10)
      COMMON/INPUT / TX(4,100)
      COMMON/POINTP/TY(4), TYP(4), TYPP(4)
      COMMON/POINTS/ NPTS, NVAR
      DIMENSION RBPANL(11), ZBPANL(11)
C
      NBODY = 1
      NBODYS = IFIX( BODYS )
      NC = XNC
      NB = XNB
      NC1 = NC - 1
      NB1 = NB - 1
      NBC = NB1*NC1
      NBCT = NC * NB1
      NCOLW = NWING * NC1
      NW2 = 2 * NWING
      NPLANE = IFIX( PLANE )
      NTHET = IFIX( BTHET )
      THKOMP = 1.0
      WING  =  FLOAT(NWING)
      SOLID =  FLOAT(ISOLID)
      PUNCH =  FLOAT(IPUNCH)
C
      CALL BODYPF( XB, RB, ZD, THETA, NBPFL, NTHET, NBODYS, XBS1, XBS2,
     1         DSB1, DSB2 )
      BODYS = FLOAT( NBODYS )
C
C
C
      CALL BODYPN
C
      REWIND 11
      WRITE (11,1020) BODYS, BTHET, PLANE, RP
      WRITE ( 11,1020) ( XB(N), N=1,NBODYS)
      WRITE ( 11,1020) ( RB(N), N=1,NBODYS)
```

```
      WRITE ( 11,1020) ( ZD(N), N=1,NBODYS)
      WRITE ( 11,1020) ( THETA(N), N=1,NTHET)
      WRITE (11,1020) XBS1,XBS2,DSB1,DSB2
      WRITE (11,1020) ( XBPANL(N), N=1,NPLANE )
      IF ( RP .GT. 0.0 ) GO TO 410
      WRITE (11,1020) ( RBPANL(N), N=1,NTHET )
      WRITE (11,1020) ( ZBPANL(N), N=1,NTHET )
  410 CONTINUE
      PERCT = 0.1
      DO 450 I = 1, NW2, 2
      WRITE (11,1020) XWI(1,I),YWI(1,I),XWI(2,I),XWI(4,I),YWI(4,I),
     1          XWI(3,I)
      WRITE (11,1020) XNB, XNC, WCASE, ZWI(1,I)
      NW1 = I / 2 + 1
      WRITE (11,1020) PIVOT(NW1), DIH(NW1)
      CALL WINGS( NW1 )
  450 CONTINUE
      TWIST = FLOAT( ITWIST )
      THICK = FLOAT( ITHICK )
      WRITE (11,1020) XMACH, SYM, THICK
      WRITE (11,1020) CKASE, CPCALC, POLAR, THICK, XNACEL, PINF, PUNCH,
     1          SOLID,WING,(SID(K),K=1,20),(IFORM(K),K=1,10)
C
      IF ( XNACEL .EQ. 0 ) GO TO 460
      CALL NACEL( XNACEL, NWING )
  460 CONTINUE

      WRITE (11,1020) ARB, DADEG
      WRITE (11,1020) ARW, TWIST
      IF ( ITWIST  .EQ. 1 ) WRITE (11,1020) ( (ARWT(J,I), J=1,NB1), I=1,
     1          NWING )
      IF ( ICOMBR  .EQ. 1 ) WRITE (11,1020) ( ( ALPHAC(I,J), I=1,NBC),
     1          J=1,NWING )
      IF ( ITHICK  .EQ. 1 ) WRITE (11,1020) ( ( ALPHAT(I,J), I=1,NBCT),
     1          J=1,NWING )
      REWIND 11
C
  805 FORMAT ( 8F10.3 )
 1020 FORMAT (7F10.4)
C
      RETURN
      END
CDECK,WINGIN
      SUBROUTINE WINGIN
C
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING,XMACHN,
     1         SYMM, KACE, NPOLAR, IRW
      COMMON/DEFAU / BODY, BODYS, BTHET, CPCALC, CASE, NRUN, DADEG, ARB,
     1          PLANE, POLAR, RP, XNACEL, SYM, THICK,  WING, XMACH,
     2          XNB, XNC, PINF, IPUNCH, SID(20), CKASE, ARW
      COMMON/WANGLE/ ALPHAC(55,10), ALPHAT(55,10), ARWT(55,10)
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1          NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2          , PIVOT(10), DIH(10)
C
      NBODY = 0
      THKOMP = 1.0
      NPLANE = IFIX( PLANE )
      PUNCH = FLOAT( IPUNCH )
      SOLID = FLOAT( ISOLID )
      WING  = FLOAT( NWING )
C
      NC = XNC
```

```
      NB = XNB
      NC1 = NC - 1
      NB1 = NB - 1
      NBC = NB1 NC1
      NBCT = NC * NB1
      NCOLW = NWING * NC1
C
C        WRITE ON TAPE 11 FOR AERO CODE
C
      REWIND  11
C
      NW2 = 2 * NWING
      DO 50 I=1,NW2,2
      WRITE (11,1020) XWI(1,I),YWI(1,I),XWI(2,I),XWI(4,I),YWI(4,I),
     1          XWI(3,I)
      WRITE (11,1020) XNB, XNC, WCASE, ZWI(1,I)
      NW1 = I / 2 + 1
      WRITE (11,1020) PIVOT(NW1), DIH(NW1)
   50 CONTINUE
C
C        READ IN WING PLAN FORM AND COMPUTE WING CAMBER, TWIST, AND
C        THICKNESS SLPOE.
C
      DO 100 I=1,NWING
      CALL WINGS(I)
  100 CONTINUE
      THICK = FLOAT( ITHICK )
      TWIST = FLOAT( ITWIST )
      WRITE (11,1020) XMACH, SYM, THICK
      WRITE (11,1020) CKASE,CPCALC,POLAR,THICK,XNACEL,PINF,PUNCH,SOLID,
     1          WING, (SID(K), K=1,20), (IFORM(K), K=1,10)
      WRITE (11,1020) ARB, DADEG
      WRITE (11,1020) ARW, TWIST
C
      IF ( ITWIST  .EQ. 1 ) WRITE (11,1020) ( (ARWT(J,I), J=1,NB1), I=1,
     1          NWING )
      IF ( ICAMBR  .EQ. 1 ) WRITE (11,1020) ( (ALPHAC(I,J), I=1,NBC),
     1          J=1,NWING )
      IF ( ITHICK  .EQ. 1 ) WRITE (11,1020) ( ( ALPHAT(I,J), I=1,NBCT),
     1          J=1,NWING )
C
      REWIND  11
C
 1020 FORMAT ( 7F10.4)
C
C
      RETURN
      END
CDECK WINGS
      SUBROUTINE WINGS( I )
C
C
C        COMPUTES WING CAMBER( ALPHAC), THICKNESS SLOPE ( ALPHAT ),
C        TWIST ( ARWT ) AND DIHEDRALS.
C
      COMMON/REPEAT/ IREPT
      COMMON/WANGLE/ ALPHAC(55,10), ALPHAT(55,10), ARWT(55,10)
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1          NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2          , PIVOT(10), DIHED(10)
      DIMENSION XC(40), YC(40), CF(115), ZU(40), ZL(40), ZCM(40)
C
      CDR = 57.295779
      NW = 2 * I - 1
      NW1 = NW + 1
```

```
      IREPT = 0
      NCH = NCHORD
      NSP = NSPAN
      NC1 = NCH - 1
      NB1 = NSPAN - 1
      NP = ( NCH-1 ) * ( NSP-1 )
      NP1 = NCH * NB1
      NPTU = NWPI(NW)
      NPTL = NWPI(NW1)
C
C
C        IFORM(I) = 0 , AIRFOIL WING, ZWI FOR UPPER AND LOWER SURFACE
C                       ARE GIVEN AT SAME LOCATIONS ( XWI, YWI )
C                 = 1 ,              ZWI ARE GIVEN AT DIFFERENT LOCALS
C                 = 2 , FLAT WING,
C
      IF ( IFORM(I)  .GT. 1 ) GO TO 450
C
C        AIRFOL WING SURFACE
C
      CALL WPANEC( XWI(1,NW), YWI(1,NW), NCH, NSP, XC, YC)
C
C        DETERMINE THE CAMBER PLANE
      IF ( IFORM(I)  .LT. 1 ) GO TO 100
C
      CALL COEFF( XWI(1,NW1), YWI(1,NW1), ZWI(1,NW1), CF, NPTL )
      CALL INTERP( NPTL, XWI(1,NW1), YWI(1,NW1), CF, NPTU, XWI(1,NW),
     1         YWI(1,NW), ZWI(1,NW1 ) )
  100 CONTINUE
      DO 200 J = 1, NPTU
      ZCM(J) = 0.5 * ( ZWI(J,NW) + ZWI(J,NW1) )
      ZU(J) = ZWI(J,NW) - ZCM(J)
  200 CONTINUE
C
C        WING  TWIST  ( ARWT )
      CALL WTWIST( YC, I )
C
C
C        WING  CAMBER  ( ALPHAC )
C
      CALL COEFF( XWI(1,NW), YWI(1,NW), ZCM, CF, NPTU )
      CALL INTERP( NPTU, XWI(1,NW),YWI(1,NW),CF,NP,XC,YC,ALPHAC(1,I) )
C
C
C        WING  THICKNESS SLOPE  ( ALPHAT )
C
      IREPT = 1
      CALL COEFF( XWI(1,NW), YWI(1, NW), ZU, CF, NPTU)
      CALL INTERP( NPTU, XWI(1,NW),YWI(1,NW),CF,NP,XC,YC,ALPHAT(1,I) )
C
C        SET ITWIST FOR WING
C
      ITWIST = 0
      DO 310 K=1,NB1
      IF ( ARWT(K,I)*CDR  .GT. 0.1 ) GO TO 315
  310 CONTINUE
      GO TO 316
  315 CONTINUE
      ITWIST = 1
  316 CONTINUE
C
C        SET ICAMBR FOR WING
C
      ICAMBR = 0
      DO 410 K=1,NP
```

```
      IF ( ALPHAC(K,I)*CDR  .GT. 0.01 ) GO TO 415
  410 CONTINUE
      GO TO 416
  415 CONTINUE
      ICAMBR = 1
  416 CONTINUE
      GO TO 500
C
  450 CONTINUE
C
C        FLAT WING SURFACE
C
      ITWIST = 0
      ICAMBR = 0
      CNX = ( YWI(2, NW) - YWI(1,NW))*(ZWI(3,NW)-ZWI(1,NW))-( YWI(3,NW)-
     1         YWI(1,NW) ) * ( ZWI(2,NW) - ZWI(1,NW) )
      CNZ = ( XWI(2, NW) - XWI(1,NW) )*( YWI(3,NW)-YWI(1,NW) )-( XWI(3,
     1         NW) - XWI(1,NW) ) *( YWI(2,NW) - YWI(1,NW) )
      ALP = ATAN( - CNX / CNZ )
      DO 470 J =1, NP
      ALPHAT(J,I) = ALP
  470 CONTINUE
C
  500 CONTINUE
C
C        REARRANGE ALPHAT TO CORRESPONDING WITH AERO CODE FORMAT
C
      DO 505 K=1, NB1
      IO = NP - K*NC1 + 1
      IR = NP1 - K*NCH + 1
      DO 504 J=1,NC1
      JO = NP - J - (K-1)*NC1 + 1
      JR = NP1 - J -(K-1)*NCH + 1
      ALPHAT(JR,I) = ALPHAT(JO,I)
  504 CONTINUE
      ALPHAT(IR,I) = ALPHAT(IO,I)
  505 CONTINUE
C
C
C        SET ITHICK FOR WING
C
      ITHICK = 0
      DO 510 K=1,NP
      IF ( ALPHAT(K,I)*CDR  .GT. 0.01 ) GO TO 515
  510 CONTINUE
      GO TO 516
  515 CONTINUE
      ITHICK = 1
  516 CONTINUE
C
      RETURN
      END
CDECK COEFF
      SUBROUTINE COEFF( X, Y, Z, CF, N )
C
C
C        COMPUTES COEFFICIENTS OF INTERPOLATION OF Z AT N POINTS(X,Y)
C
      DIMENSION X(1), Y(1), Z(1), CF(1), XT(115), YT(115), CFT(115)
C
      N3 = N + 3
      DO 100 I=1,N
      XT(I) = X(I)
      YT(I) = Y(I)
      CFT(I) = Z(I)
```

```
  100 CONTINUE
C
      CALL SURFIT( XT, YT, N3, CFT )
C
      DO 200 I=1,N3
      CF(I) = CFT(I)
  200 CONTINUE
C
      RETURN
      END
CDECK INTERP
      SUBROUTINE INTERP( N, X, Y, CF, NP, XP, YP, ZP )
C
C
C        INTERPOLATES AT NP POINTS ( XP, YP )
C
      DIMENSION X(1), Y(1), XP(1), YP(1), ZP(1), CF(1)
C
      DO 100 I=1,NP
      SUM = CF(2)
      DO 110 J=1,N
      J3 = J + 3
      RI = (XP(I) - X(J) )**2 + ( YP(I) - Y(J) )**2
      IF ( RI  .LT. 1.E-9 ) GO TO 110
      SUM = SUM+ 2.0*CF(J3)*(1.0+ALOG(RI) )*(XP(I)-X(J) )
  110 CONTINUE
      ZP(I) = SUM
  100 CONTINUE
C
      RETURN
      END
CDECK WPANEC
      SUBROUTINE WPANEC( XI, YI, NC, NB, XC, YC )
C
C
C        COMPUTE CONTROL POINT ( XC, YC ) OF WING PANELS
C
      DIMENSION XI(1), YI(1), XC(1), YC(1), XY(13,11), XW(13), YW(11),
     1          DY(11)
C
      PERCT = 0.1
      XCPT = 0.95
      XW(1) = XI(1)
      XCHORD = XI(2) - XI(1)
      XW(2) = XW(1) + PERCT*XCHORD
      DX = (1.0 - PERCT) * XCHORD / ( NC - 2 )
      DO 100 L=3,NC
      XW(L) = XW(L-1) + DX
  100 CONTINUE
      BY = YI(4) - YI(1)
      DBY = BY / ( NB - 1)
      YW(1) = YI(1)
      DO 165 N=2,NB
      YW(N) = YW(N-1) + DBY
  165 CONTINUE
      SLE = ( XI(4)-XI(1))/ BY
      STE = (XI(3)-XI(2) )/ BY
      IF ( SLE-STE) 210, 230, 210
  210 CONTINUE
      DX = ( XI(3) - XI(4) )/ ( XI(2) - XI(1) )
      DO 215 M=1,NC
      XY( M, NB ) = XI(4) + DX * ( XW(M) - XI( 1 ) )
  215 CONTINUE
      DO 220 N=1,NB
      DY(N) = ( YW(N) - YI(1) ) / BY
```

```
      DO 220 M=1,NC
      XY(M, N) = XW(M) + ( XY( M, NB) - XW(M) ) * DY(N)
  220 CONTINUE
      GO TO 250
  230 CONTINUE
      DO 240 N=1,NB
      DY(N) = YW(N) - YI(1)
      DO 240 M=1,NC
      XY( M, N) = XW(M) + SLE* DY(N)
  240 CONTINUE
  250 CONTINUE
C
C        CONTROL POINT COORDINATES ( XC, YC)
C
      NC1 = NC-1
      NB1 = NB-1
      DO 350 N=1,NB1
      DO 350 M=1,NC1
      K = M + (N-1)*NC1
      CT = XY(M+1,N+1) - XY(M,N+1)
      CR = XY(M+1,N) - XY(M,N)
      RY = ( 1.0 +CT/(CR+CT) )/3.0
      RY1 = 1.0 - RY
      XWL = RY*XY(M, N+1) + RY1*XY(M, N)
      XWT = RY * XY(M+1, N+1) + RY1*XY(M+1, N)
      XC(K) = XCPT * XWT + ( 1.0 - XCPT) * XWL
      YC(K) = YW( N+ 1) * RY + YW(N) * RY1
  350 CONTINUE
C
      RETURN
      END
CDECK WTWIST
      SUBROUTINE WTWIST( YC, I )
C
C
C        COMPUTES THE WING TWIST ANGLE  ARWT
C
      COMMON/WANGLE/ ALPHAC(55,10), ALPHAT(55,10), ARWT(55,10)
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1          NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2            , PIVOT(10), DIH(10)
      DIMENSION YC(1), VX(2), VY(2), VZ(2)
C
C     COMPUTES LEADING AND TRAILING EDGE UNIT VECTORS
C
      DO 100 J=1,2
      J1 = 4
      J2 = 1
      IF ( J .EQ. 2 ) J1 = 3
      IF ( J .EQ. 2 ) J2 = 2
      DX = XWI(J1,I) - XWI(J2,I)
      DY = YWI(J1,I) - YWI(J2,I)
      DZ = ZWI(J1,I) - ZWI(J2,I)
      SUML = SQRT( DX**2 + DY**2 + DZ**2 )
      VX(J) = DX / SUML
      VY(J) = DY / SUML
      VZ(J) = DZ / SUML
  100 CONTINUE
C
      NB1 = NSPAN - 1
      NC1 = NCHORD - 1
      DO 200 K=1,NB1
      K1 = 1 + ( K-1 ) * NCHORD
      DY = YC( K1 ) - YWI( 1, I )
      XLE = XWI(1,I) + VX(1) * DY / VY(1)
```

```
      XTE = XWI(2,I) + VX(2) * DY / VY(2)
      ZLE = ZWI(1,I) + VZ(1) * DY / VY(1)
      ZTE = ZWI(2,I) + VZ(2) * DY / VY(2)
      ARWT(K,I) = ( ZLE-ZTE ) / ( XTE-XLE )
  200 CONTINUE
C
      RETURN
      END
      SUBROUTINE WTRANS( NW, IW )
C
C
C        TRANSFER COORDINATES DEFINING WING FORM INTO THAT OF AERO CODE
C
      COMMON/WINGFM/ XWI(30,20), YWI(30,20), ZWI(30,20),NWPI(20),NCHORD,
     1         NSPAN, ISOLID, NPLANE, ICAMBR, ITWIST,ITHICK,IFORM(10)
     2          , PIVOT(10), DIH(10)
      COMMON/CORTRN/ NSYM, NTRANS(20), ISYM(20), VA(3,2,20),BTFM(9,2,20)
C
      DR = 0.0174532925
      NWP = NWPI( NW )
      IF ( IFORM(IW)  .EQ. 2 ) NWP = 4
      DO 400 J=1,NWP
      R = XWI( J, NW )
      THETA = YWI( J, NW) * DR
      PHI = ZWI( J, NW) * DR
C
      NGO = ISYM( NSYM ) + 1
      GO TO ( 300, 100, 200) , NGO
  100 CONTINUE
      XWI( J, NW) = R * COS( THETA )
      YWI( J, NW) = R * SIN( THETA )
      GO TO 300
C
  200 CONTINUE
      XWI(J,NW) = R * SIN( THETA ) * COS( PHI )
      YWI(J,NW) = R * SIN( THETA ) * SIN( PHI )
      ZWI(J,NW) = R * COS( THETA )
C
  300 CONTINUE
C
  400 CONTINUE
C
      RETURN
      END
CDECK NACEL
      SUBROUTINE NACEL( XNACEL, NWING )
C
C
C        INTERPOLATES INPUT OF NACEL CONFIGURATION AND WRITES ON TAPE 11
C        FOR AERO CODE.
C
      COMMON/ BODY2 / NBPFL, NBODYS, NTHET, XB(51), RB(51), ZD(51),
     1         THETA(11), RP,  XBPANL(20)
      COMMON/BODYSP/ DRDX(51), DZDX(51)
      COMMON/ NACEL /NACELP, XN(51,2), RN(51,2), ZN(51,2), ARN(2)
      COMMON/CORTRN/ NSYM, NTRANS(20), ISYM(20), VA(3,2,20),BTFM(9,2,20)
C
      ANCEL = ABS( XNACEL )
      NACELS= IFIX(ANCEL)
      NW2 = 2 * NWING
      DO 300 N = 1, NACELS
      CALL BODYPF( XN(1,N), RN(1,N), ZN(1,N), THETA, NACELP, NBODYS, XN1
     1          , XN2, DS1, DS2)
      NSYM = 1 + NW2 + N
      YI = VA( 2, 1, NSYM ) + VA( 2, 2, NSYM )
```

```
      ZI = VA( 3, 1, NSYM ) + VA( 3, 2, NSYM )
C
      WRITE (11,1020) XN(1,N), YI, ZI, NBODYS, NTHET, ARN(N)
      IF ( XNACEL) 100, 300, 200
C
  100 CONTINUE
      IF ( N - 1 ) 300, 200, 300
  200 CONTINUE
      WRITE (11,1020) ( XN(L,N), L=1,NBODYS )
      WRITE (11,1020) ( RN(L,N), L=1,NBODYS )
      WRITE (11,1020) ( ZN(L,N), L=1,NBODYS )
      WRITE (11,1020) ( THETA(L), L=1,NTHET )
      WRITE (11,1020)  XN1, XN2, DS1, DS2,(DRDX(J),DZDX(J), J=1,NBODYS)
  300 CONTINUE
C
 1020 FORMAT ( 7F10.4 )
C
      RETURN
      END
      SUBROUTINE TRANSM( A, B, C, UNIVEC)
C
C
C              COMPUTES THE TRANSFORM MATRIX TFM BETWEEN TWO COORDINATE
C              SYSTEMS 1 AND 2 . A  DEFINE THE ORIGIN,  B  A POINT ON
C              THE Z AXIS AND C  A POINT ON THE X - Z PLANE OF COORDINA-
C              TE SYSTEM 2.
C         THE MATRIX UNIVEC(L,M) HAS THE ROW INDEX L FOR THE UNIT VECTOR
C           I, J, K AND THE COLUMN INDEX M FOR THEIR THREE COMPONENTS IN
C           THE COORDINATE SYSTEM  1 .
C
      DIMENSION A(3), B(3), C(3), UNIVEC(3,3), UNIVCA( 3 ), N1(3), N2(3)
      DATA N1 / 3, 1, 2 /, N2 / 2, 3, 1 /
C
C         COMPUTE THE UNIT VECTOR  I, J, K OF COORDINATE SYSTEM 2 IN TERM
C           OF COORDINATE SYSTEM 1 .
C
C
C
      SUM = 0.0
      SUM1 = 0.0
      DO 100 I = 1, 3
      SUM = SUM + ( B(I) - A(I) )**2
      SUM1 = SUM1 + ( C(I) - A(I) )**2
  100 CONTINUE
      AB = SQRT ( SUM )
      CA = SQRT( SUM1 )
C
C            UNIT  VECTOR  K
C              UNIT VECTOR  CA
C
      DO 200 J= 1, 3
      UNIVEC(3, J) = ( B(J) - A(J) )/ AB
      UNIVCA( J ) = ( C(J) - A(J) ) / CA
  200 CONTINUE
C
C            UNIT  VECTOR  J
C
      SUM = 0.0
      DO 300 I = 1, 3
      I1 = N1(I)
      I2 = N2(I)
      UNIVEC(2,I) = UNIVCA(I1)*UNIVEC(3,I2)-UNIVCA(I2)*UNIVEC(3,I1)
      SUM = SUM + UNIVEC( 2, I)**2
  300 CONTINUE
      SUM = SQRT( SUM )
```

```
      DO 350 I=1,3
      UNIVEC(2,I) = UNIVEC(2,I) / SUM
  350 CONTINUE
C
C            UNIT  VECTOR  I
C
      SUM = 0.0
      DO 400 I = 1, 3
      I1 = N1(I)
      I2 = N2(I)
      UNIVEC( 1, I) = UNIVEC(2, I2) * UNIVEC( 3, I1 ) - UNIVEC( 2, I1 )*
     1          UNIVEC( 3, I2 )
      SUM = SUM + UNIVEC( 1, I)**2
  400 CONTINUE
      SUM = SQRT( SUM )
      DO 450 I=1,3
      UNIVEC(1,I) = UNIVEC(1,I) / SUM
  450 CONTINUE
C
      RETURN
      END
CDECK,CUBERT
      SUBROUTINE CUBERT
      DIMENSION X(3)
      COMMON/LOCTN / V(3), L, M, G(4), VMAX, VMIN
      COMMON/POINTS/ NPTS, NVAR
      L=0
      SUM = SQRT( G(1)**2+G(2)**2+G(3)**2+G(4)**2 )
      G1 = G(1) / SUM
      G2 = G(2) / SUM
      G3 = G(3) / SUM
      G4 = G(4) / SUM
C
C        EQ. 1    G1*X**3 + G2*X**2 + G3*X = 0.
      IF ( ABS( G4) .LT. 1.E-6) GO TO 4
C
C        EQ. 2    G1*X**3 + G2*X**2 + G3*X +G4 = 0.
      IF ( ABS( G1) .GT. 1.E-6) GO TO 9
C
C        EQ. 3    G2*X**2 + G3*X + G4 = 0.
      IF ( ABS( G2) .GT. 1.E-6) GO TO 7
C
C        EQ. 4    G3*X + G4 = 0.
      IF ( ABS( G3) .GT. 1.E-6) GO TO 5
      RETURN
C
    4 CONTINUE
      V(1) = 0.0
      L=1
      RETURN
    5 CONTINUE
      V(1) = - G(4) / G(3)
      GO TO 35
    7 CONTINUE
      R1 = G(3)**2 - 4.0*G(2)*G(4)
      IF (  R1  .GT. 0.0 )  GO TO 107
      WRITE (6,901)
  901 FORMAT (1H0,*NAGATIVE RADICAL RESULTED FROM SOLVING THE QUADRATIC
     1EQUATION  G2.X2 + G3.X + G4 = 0*)
      RETURN
  107 CONTINUE
      R1=SQRT(R1)
      V1 = ( -G(3) + R1 ) *0.5 / G(2)
      V(1) = V1
      IF ( V1  .GT. VMIN .AND.  V1  .LT. VMAX ) GO TO 8
```

```
      V2 = (-G(3) - R1 )*0.5 / G(2)
      V(2) = V2
      IF ( V2  .GT. VMIN .AND.  V2  .LT. VMAX ) GO TO 8
      WRITE (6,902) VMIN, V1,V2, VMAX
  902 FORMAT (1H0,*BOTH ROOTS V1, V2 OF QUADRATIC EQ. G2.X2 + G3.X + G4
     1= 0 OUTSIDE OF LIMIT VMIN, VMAX*/ 1X,*VMIN =*,E14.4,4X,*V1 =*,
     2          E14.4,4X,*V2 =*,E14.4,4X,*VMAX =*,E14.4)
      RETURN
    8 CONTINUE
      L = 1
      RETURN
C
C         THREE ROOTS
C
    9 CONTINUE
      A1 = G(2) / G(1)
      A2=G(3)/G(1)
      A3=G(4)/G(1)
      A=A2-A1*A1/3.
      Q=2.*A1*A1*A1/27.-A1*A2/3.+A3
      RAD=Q*Q/4.+A*A*A/27.
      SIGN=1.
      IF(Q.GE.0.)SIGN=-1.
      IF(RAD)10,20,20
C
C               THREE UNEQUAL REAL ROOTS
C
   10 CONTINUE
      A3 = SQRT( ABS( A/3.0) )
      A2 = SQRT( ABS( A**3/3. ) )
      PHI = ACOS( ABS( 1.5*Q/ A2) )
      DO 11 I=1,3
      X(I) = 2.0*SIGN*A3*COS((PHI+6.2831853*FLOAT(I-1))/3.0)
   11 CONTINUE
      GO TO 40
C
C         EITHER ONE REAL AND TWO IMAG. OR THREE REAL ROOTS OF WHICH TWO
C         ARE EQUAL
C
   20 CONTINUE
      V1 = 1.0 / 3.0
      Q = A2*V1 - ( A1*V1)**2
      R = .5*A1*A2*V1 - .5 *A3 - ( A1*V1) **3
      RAD = Q**3 + R*R
      IF ( RAD  .GT.  0.0 )  GO TO 120
      WRITE (6,903)
  903 FORMAT (1H0,*ERROR IN TESTING RADICAL, REDUNDENT TEST*)
      RETURN
  120 CONTINUE
      RAD = SQRT( RAD)
      A2 = R + RAD
      SIGN = 1.
      IF ( A2  .LT.  0.0)  SIGN = -1.
      A2 = SIGN * ABS( A2 ) **V1
      A3 = R - RAD
      SIGN = 1.
      IF ( A3 .LT. 0.0 ) SIGN = -1.
      A3 = SIGN * ABS( A3 )**V1
      V0 = A2 + A3 - A1*V1
   35 CONTINUE
      IF ( V0 .GT. VMIN .AND. V0  .LT. VMAX ) L = 1
      IF ( L  .EQ. 0 ) WRITE (6,905)
  905 FORMAT (1H0,*THE ONLY UNEQUAL ROOT IS OUTSIDE OF VMIN, VMAX*/1X,
     1          *VMIN =*,E14.4,4X,* V =*,E14.4,4X,*VMAX =*,E14.4)
      V(1) = V0
```

```
      RETURN
   40 CONTINUE
      DO 45 I=1,3
      VO = X(I) - A1/3.0
      X(I) = VO
      V(I) = VO
      IF ( VO .LT. VMIN .OR.  VO  .GT. VMAX ) GO TO 50
      L=1
      RETURN
   50 CONTINUE
   45 CONTINUE
      WRITE (6,904) VMIN, V(1), V(2), V(3), VMAX
  904 FORMAT (1HO,*ALL THREE UNEQUAL ROOTS OUTSIDE OF LIMIT VMIN, VMAX*/
     1    1X,*VMIN =*,E14.4,4X,*V1 =*,E14.4,4X,*V2 =*,E14.4,4X,*V3 =*,
     2        E14.4,4X,*VMAX =*,E14.4)
C
      RETURN
      END
CDECK,POINT
      SUBROUTINE POINT( N )
      COMMON/ CURFIT/ B( 4, 4, 100), T(100), SCALEF( 4 )
      COMMON/ ENDPT/ XFIRST, XLAST, XSAVE(100)
      COMMON/LOCTN/ V(3), L, M, A(4), VMAX, VMIN
      COMMON/POINTS/ NPTS, NVAR
      COMMON/ POINTP/  X(4),  XP( 4),  XPP( 4)
      DIMENSION DX(4)
C
      N1 = NPTS - 1
      IF(N.NE.0)GO TO 4
      DO 1 I=1,N1
      M=I
      IF ( V(1)  .LE. T(I+1) ) GO TO 2
    1 CONTINUE
      WRITE(6,100)
  100 FORMAT(1HO,33H ERROR - PARAMETER EXCEEDS LIMITS)
      RETURN
    2 CONTINUE
      DO 300 K=1,3
      P = V(K)
      DX(K) = 1.E10
      IF ( P .LT. VMIN  .OR.  P  .GT. VMAX ) GO TO 300
      DX(K) = ABS((((B(1,1,M)*P + B(1,2,M))*P + B(1,3,M))*P + B(1,4,M))*
     1          SCALEF(1) - XIND )
  300 CONTINUE
      IF ( DX(1)  .GT. DX(2) ) GO TO 302
      IDENT = 1
      IF ( DX(1)  .GT. DX(3) )  IDENT = 3
      GO TO 305
  302 CONTINUE
      IDENT = 2
      IF ( DX(2)  .GT. DX(3) )  IDENT = 3
  305 CONTINUE
      P = V(IDENT)
C
      DO 3   I=1, NVAR
      X(I)=(((B(I,1,M)*P+B(I,2,M))*P+B(I,3,M))*P+B(I,4,M))*SCALEF(I)
      XP(I)=((3.*B(I,1,M)*P+2.*B(I,2,M))*P+B(I,3,M))*SCALEF(I)
    3 XPP(I)=(6.*B(I,1,M)*P+2.*B(I,2,M))*SCALEF(I)
C
  310 CONTINUE
      RETURN
C
    4 X(N)=X(N)/SCALEF(N)
      DO 5 I=1,N1
      M=I
```

```
      K = I + 1
      IF ( XSAVE(K)  .GE. X(N) ) GO TO 6
    5 CONTINUE
      XIND = X(N) * SCALEF(N)
      XFT = XFIRST * SCALEF(1)
      XLT = XLAST * SCALEF(1)
      WRITE (6,101)  XIND, XFT, XLT
  101 FORMAT (1H0,*  THE GIVEN VALUE  F INDEPENDENT VARIABLE *,E12.4,
     1         5X,*IS OUT OF RANGE TO BE INTERPOLATED   *,E12.4,5X,E12.4)
      RETURN
    6 DO 7 I=1,4
    7 A(I)=B(N,I,M)
      A(4)=A(4)-X(N)
      IF ( XFIRST   .EQ. X(N) ) GO TO 112
      IF ( XLAST    .EQ. X(1) ) GO TO 111
      VMAX = T(M+1) * 1.0001
      VMIN = T(M) * 0.9999
      CALL CUBERT
      XIND = X(1)* SCALEF(1)
      GO TO 103
  112 V(1) = 0.0
      L=1
      GO TO 103
  111 V(1) = 1.0
      L=1
  103 CONTINUE
      IF(L.NE.0)GO TO 2
      OP = P
      WRITE (6,10) OP
   10 FORMAT (1X4HOP =,E15.8)
      DO 8 J=1,2
      K = M + J - 1
      B1 = B(N,1,K)
      B2 = B(N,2,K)
      B3 = B(N,3,K)
      B4 = B(N,4,K)
      P = T(K)
      SUM = B1*P**3 + B2*P**2 + B3*P + B4
      DIF = SUM - X(N)
    8 WRITE (6,9) K,P,B1,B2,B3,B4,X(N),SUM,DIF
    9 FORMAT (1XI3,8E15.8)
      WRITE(6,102)
  102 FORMAT(1H0,32H FAILURE IN CUBE ROOT EXTRACTION)
      RETURN
      END
CDECK,SPLFIT
      SUBROUTINE SPLFIT
      REAL L,M,MU
      COMMON/CURFIT/ B(4,4,100), T(100), SCALEF(4)
      COMMON/ ENDPT/ XFIRST, XLAST, XSAVE(100)
      COMMON/INPUT /  X( 4, 100)
      COMMON/POINTS/ NPTS, NVAR
      DIMENSION M(4,100), S(100), L(100), MU(100), D(100), P(100), Q(100
     1          ), U(100)
      IF(NPTS.GT.100)NPTS=100
      DO 200 I=1, NVAR
  200 SCALEF(I)=0.
      DO 201 I=1,NPTS
      DO 201 J=1,NVAR
  201 SCALEF(J)=AMAX1(SCALEF(J),ABS(X(J,I)))
      DO 101 J=1,NVAR
      IF ( SCALEF(J)  .LT. 1.E-10 ) SCALEF(J) = 1.0
  101 CONTINUE
      DO 202 I=1,NPTS
      DO 202 J=1,NVAR
```

```
202 X(J,I)=X(J,I)/SCALEF(J)
    DO 203 I=1,NPTS
    XSAVE(I) = X(1,I)
203  CONTINUE
    XFIRST = X(1,1)
    XLAST = X(1,NPTS)
    T(1)=0.
    S(1)=0.
    SUM=0.
    L(1)=-2.
    MU(1)=0.
    L(NPTS)=0.
    MU(NPTS)=-2.
    D(1)=0.
    D(NPTS)=0.
    P(1)=2.
    Q(1)=1.
    U(1)=0.
    N1=NPTS-1
    DO 2 I=2,NPTS
    SUM1=0.
    DO 1 J=1,NVAR
  1 SUM1=SUM1+(X(J,I)-X(J,I-1))**2
    S(I)=SQRT(SUM1)
  2 T(I)=T(I-1)+S(I)
    DO 3 I=2,NPTS
    S(I)=S(I)/T(NPTS)
  3 T(I)=T(I)/T(NPTS)
    DO 4 I=2,N1
    L(I)=S(I+1)/(S(I)+S(I+1))
    MU(I)=1.-L(I)
    P(I)=MU(I)*Q(I-1)+2.
  4 Q(I)=-L(I)/P(I)
    P(NPTS)=2.*(1.-Q(N1))
    DO 7 I=1,NVAR
    DO 5 J=2,N1
    D(J)=6.*((X(I,J+1)-X(I,J))/S(J+1)-(X(I,J)-X(I,J-1))/S(J))/(S(J)+
   1S(J+1))
  5 U(J)=(D(J)-MU(J)*U(J-1))/P(J)
    U(NPTS)=-MU(NPTS)*U(N1)/P(NPTS)
    M(I,NPTS)=U(NPTS)
    DO 6 J=1,N1
    K=NPTS-J
    M(I,K)=Q(K)*M(I,K +1)+U(K)
    B(I,1,K)=(M(I,K +1)-M(I,K))/(6.*S(K+1))
    B(I,2,K)=.5*(M(I,K)*T(K+1)-M(I,K+1)*T(K))/S(K+1)
    B(I,3,K)=(.5*(M(I,K+1)*T(K)**2-M(I,K)*T(K+1)**2)+X(I,K+1)-X(I,K)+
   1(M(I,K)-M(I,K+1))*S(K+1)**2/6.)/S(K+1)
  6 B(I,4,K)=(M(I,K)*T(K+1)**3-M(I,K+1)*T(K)**3)/(6.*S(K+1))+(X(I,K)*T
   1(K+1)-X(I,K+1)*T(K))/S(K+1)+S(K+1)/6.*(M(I,K+1)*T(K)-M(I,K)*T(K+1)
   1)
  7 CONTINUE
    RETURN
    END
CDECK INVERT
      SUBROUTINE INVERT (GOTON, A, IA, IPIVOT, INDXR, INDXC, NROWS)
C                                                                       INVE1010
C ..................................................................... INVE1020
C                                                                       INVE1030
C ..................................................................... INVE1040
C                                                                       INVE1050
      REAL A(NROWS, NROWS), PIVOT, T                                    INVE1060
      INTEGER IPIVOT(IA), INDXR(IA), INDXC(IA)                          INVE1070
C                                                                       INVE1080
C                                                                       INVE1090
```

```
      GOTON = 1
      N = IA                                                            INVE1100
      DO 1000 J=1, N                                                    INVE1110
 1000 IPIVOT(J) = 0                                                     INVE1120
      DO 1090 I=1, N                                                    INVE1130
      T = 0.0                                                           INVE1140
      DO 1020 J=1, N                                                    INVE1150
      IF (IPIVOT(J).EQ.1) GO TO 1020                                    INVE1160
C                                                                       INVE1170
      DO 1010 K=1, N                                                    INVE1180
      IF (IPIVOT(K).EQ.1) GO TO 1010                                    INVE1190
      IF ( .NOT. (ABS(A(J,K)) -ABS(T) .GT. 0.0) ) GO TO 1010            INVE1200
      IROW = J                                                          INVE1210
      ICOL = K                                                          INVE1220
      T = A(J,K)                                                        INVE1230
 1010 CONTINUE                                                          INVE1240
 1020 CONTINUE                                                          INVE1250
      IPIVOT(ICOL) = IPIVOT(ICOL)+1                                     INVE1260
      IF (IROW.EQ.ICOL) GO TO 1040                                      INVE1270
      DO 1030 L=1, N                                                    INVE1280
      T = A(IROW,L)                                                     INVE1290
      A(IROW,L) = A(ICOL,L)                                             INVE1300
 1030 A(ICOL,L) = T                                                     INVE1310
 1040 INDXR(I) = IROW                                                   INVE1320
      INDXC(I) = ICOL                                                   INVE1330
      PIVOT = A(ICOL,ICOL)                                              INVE1340
      IF (PIVOT)1050,1120,1050                                          INVE1350
 1050 A(ICOL,ICOL) = 1.0                                                INVE1360
      DO 1060 L=1, N                                                    INVE1370
 1060 A(ICOL,L) = A(ICOL,L)/PIVOT                                       INVE1380
      DO 1080 L=1, N                                                    INVE1390
      IF (L.EQ.ICOL) GO TO 1080                                         INVE1400
      T = A(L,ICOL)                                                     INVE1410
      A(L,ICOL) = 0.0                                                   INVE1420
      DO 1070 M=1, N                                                    INVE1430
 1070 A(L,M) = A(L,M)-A(ICOL,M)*T                                       INVE1440
 1080 CONTINUE                                                          INVE1450
 1090 CONTINUE                                                          INVE1460
      DO 1110 I=1, N                                                    INVE1470
      L = N+1-I                                                         INVE1480
      IF( INDXR(L) .EQ.  INDXC(L) )  GO TO 1110                         INVE1490
      IROW = INDXR(L)                                                   INVE1500
      ICOL = INDXC(L)                                                   INVE1510
      DO 1100 K=1, N                                                    INVE1520
      T = A(K,IROW)                                                     INVE1530
      A(K,IROW) = A(K,ICOL)                                             INVE1540
 1100 A(K,ICOL) = T                                                     INVE1550
 1110 CONTINUE                                                          INVE1560
C                                                                       INVE1570
C                                                                       INVE1580
C     SUCCESSFUL SOLUTION                                               INVE1590
C                                                                       INVE1600
C -----------------                                                     INVE1610
      RETURN                                                            INVE1620
C -----------------                                                     INVE1630
C                                                                       INVE1640
 1120 CONTINUE                                                          INVE1650
C                                                                       INVE1660
C     SINGULAR MATRIX                                                   INVE1670
      WRITE (6,1130)                                                    INVE1680
C                                                                       INVE1690
C -----------------                                                     INVE1700
      GOTON = 2
      RETURN
C -----------------                                                     INVE1720
```

```
 1130 FORMAT (29H ERROR THE MATRIX IS SINGULAR)                  INVE1730
      END
CDECK,SURFIT
      SUBROUTINE  SURFIT( X, Y, N3, B )
C
C          COMPUTES COEFFICIENTS FOR SURFACE SPLINE.
C          N  TOTAL NUMBER OF INPUT POINTS, CAN BE RANDOMLY LOCATED.
C          JEQ  FOR EQUATION NUMBERING
C          MUK  FOR COEFFICIENT NUMBERING
C
C          CF  ARE THE COEFFICINT MATRIX OF EQUATION
C                W(X,Y) = B(1) + B(2)*X + B(3)*Y + SUM OF B(I) * R(I)**2 *
C                LN( R(I)**2)
C
      COMMON/REPEAT/ IREPT
      DIMENSION IPIV(115), INXR(115), INXC(115), CF(115,115), TM(115)
      DIMENSION X(N3), Y(N3), B(N3)
C
      N4 = 115
      N2 = N3 - 1
      N1 = N3 - 2
      N = N3 - 3
C
C          BYPASS MATRIX INVERSION FOR THE CASE WHERE THE DATA POINTS (X,
C          Y )  ARE IDENTICAL FOR INTERPOLATION
C
C
  101 CONTINUE
      DO 100 JEQ = 1, N
      CF( JEQ, 1) = 1.0
      CF ( JEQ, 2 ) =    X( JEQ )
      CF ( JEQ, 3 ) =    Y( JEQ )
      DO 90 MUK = 1, N
      MUK1 = MUK + 3
      IF ( MUK  .EQ.  JEQ ) GO TO 90
      RJM = ( X( MUK ) - X( JEQ ) )**2 + ( Y( MUK ) - Y( JEQ ) )**2
      CF( JEQ, MUK1 ) = RJM * ALOG( RJM )
   90 CONTINUE
  100 CONTINUE
C
      DO 150 K=1,3
      CF( N1, K) = 0.0
      CF( N2, K) = 0.0
      CF( N3, K) = 0.0
  150 CONTINUE
      DO 200  MUK =1,N
      MUK1 = MUK + 3
      CF( N1, MUK1) = 1.0
      CF ( N2, MUK1) =    X( MUK )
      CF( N3, MUK1 ) =    Y( MUK )
  200 CONTINUE
C
C
      IA = N3
      CALL INVERT( GOT, CF, IA, IPIV, INXR, INXC, N4)
C
  250 CONTINUE
      B(N1) = B(N2) = B(N3) = 0.0
C
      DO 300 I=1,N3
      SUM = 0.0
      DO 290 J=1,N3
      SUM = SUM + CF(I, J) * B(J)
  290 CONTINUE
      TM(I) = SUM
```

```
  300 CONTINUE
C
      DO 400 I=1,N3
      B(I) = TM(I)
  400 CONTINUE
C
      RETURN
      END
CDECK AMATE
      OVERLAY(WOOD,3,0)
      PROGRAM AMATE
C                                                                       AMAT1010
C ..................................................................... AMAT1020
C     CONTROL ROUTINE FOR COMPUTING AERODYNAMIC INFLUENCE COEFFICIENTS  AMAT1030
C     MATRIX AND VELOCITY COMPONENTS                                    AMAT1040
C --------------------------------------------------------------------- AMAT1050
C                                                                       AMAT1060
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /BLOCK/ALPHAS(210),AREA(210),A(210),ALPHAC(110),ALPHAT(110)AMAT1090
     1,CHORD(210)                                                       AMAT1100
     2,ISYM                                                             AMAT1110
     3,NPART(210),NPANEL,NROW(2)                                        AMAT1120
     4,THETA(210),TAIL,THKOMP                                           AMAT1130
     5,U(210)                                                           AMAT1140
     6,V(210),VPM(210),VV(210),VPMM(210)                                AMAT1150
     7,W(210),WPM(210),WW(210),WPMM(210)                                AMAT1160
     8,X(210,3,4),XBAR(210),XC(210)                                     AMAT1170
     9,Y(210,3,4),YBAR(210),YC(210)                                     AMAT1180
     A,Z(210,3,4),ZBAR(210),ZC(210)                                     AMAT1190
C                                                                       AMAT1200
C     READ IN GEOMETRICAL DATA                                          AMAT1210
      CALL INTAPE                                                       AMAT1220
      WRITE(NTAPEO,1040)                                                AMAT1230
C                                                                       AMAT1240
C     KACE = 1      WING ALONE                                          AMAT1250
C     KACE = 2      BODY ALONE                                          AMAT1260
C     KACE = 3      WING-BODY COMBINATION                               AMAT1270
C                                                                       AMAT1280
      GO TO (1000,1030,1000),KACE                                       AMAT1290
C                                                                       AMAT1300
C     COMPUTE VELOCITY COMPONENTS DUE TO WING SOURCES                   AMAT1310
C     THICKNESS CASE                                                    AMAT1320
 1000 IF (THKOMP .EQ. 0.0) GO TO 1010                                   AMAT1330
      NS=NBODY+1                                                        AMAT1340
      CALL EVAL(NS,XBAR,1.)                                             AMAT1350
      WRITE(NTAPEO,1050)                                                AMAT1360
      GO TO 1020                                                        AMAT1370
 1010 CALL FOREOF (NTAPEA)                                              AMAT1380
      CALL FOREOF (NTAPEB)                                              AMAT1390
C     COMPUTE VELOCITY COMPONENTS TO WING PANEL SINGULARITIES           AMAT1400
C     LIFTING CASE                                                      AMAT1410
 1020 NS=1                                                              AMAT1420
      CALL EVAL (NS,XC,0.)                                              AMAT1430
      WRITE(NTAPEO,1050)                                                AMAT1440
 1030 REWIND NTAPEC                                                     AMAT1450
      REWIND NTAPEA                                                     AMAT1460
      REWIND NTAPEB                                                     AMAT1470
C                                                                       AMAT1480
C ------------------                                                    AMAT1490
C ------------------                                                    AMAT1510
C                                                                       AMAT1520
C                                                                       AMAT1530
 1040 FORMAT(1H0,11HEXIT INTAPE)                                        AMAT1540
 1050 FORMAT(1H0,9HEXIT EVAL)                                           AMAT1550
```

```
      END
CDECK EVAL
      SUBROUTINE EVAL (NI,XCPT,THKW)
C
C ...................................................................
C     COMPUTES AERODYNAMIC INFLUENCE COEFFICIENTS MATRIX AND VELOCITY
C     COMPONENTS MATRIX
C ...................................................................
C
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1         NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /PANVEL/ VS1(110),VS2(110),WSS(110)
      COMMON /BLOCK/ALPHAS(210),AREA(210),A(210),ALPHAC(110),ALPHAT(110)
     1,CHORD(210)
     2,ISYM
     3,NPART(210),NPANEL,NROW(2)
     4,THETA(210),TAIL,THKOMP
     5,U(210)
     6,V(210),VPM(210),VV(210),VPMM(210)
     7,W(210),WPM(210),WW(210),WPMM(210)
     8,X(210,3,4),XBAR(210),XC(210)
     9,Y(210,3,4),YBAR(210),YC(210)
     A,Z(210,3,4),ZBAR(210),ZC(210)
C
      DIMENSION AS(210),BBETAM(4),C(4),CPM(4),CPMM(4),Q(4)
     1,R(4),RPM(4),RPMM(4),UU(210),US(210),VS(210),WS(210)
     2,XCPT(210)
C
      ASINH(B)=ALOG(B+SQRT(1.+B*B))
      ACOSH(E)=2.*ALOG(SQRT(ABS((E+1.)/2.))+SQRT(ABS((E-1.)/2.)))
C
C
      EPS=0.0001
      BETA=SQRT(ABS(XMACH**2-1.))
      PI=3.1415926
      CONSTA=BETA/4.0
      CONSTB=CONSTA/PI
      CONSTD=BETA/PI
      LVSD=THKW
      IT=1
C
C     INFLUENCING PANEL I
C
      GO TO (1000,2580,1010),KACE
 1000 NC=NWING/NROW(1)
      NR=NROW(1)
      I1=1
      I2=NWING
      GO TO 1050
 1010 IF(THKW)1020,1040,1020
 1020 NC=NWING/NROW(2)
      NR=NROW(2)
      NCT=NI-1
      I1=NI
      I2=NPANEL
      DO 1030 N=1,NBODY
 1030 XCPT(N)=XC(N)
      GO TO 1060
 1040 NC=NBODY/NROW(1)
      NR=NROW(1)
      I1=1
      I2=NBODY
 1050 NCT=0
 1060 N=1
```

```
C
C
C ******************
      DO 2530 I=I1,I2
C ******************
 1080 IF(LVSD)1090,1130,1090
 1090 SL=1.0
      IT=1+I-NI-NR*(N-1)
      IF(IT)1110,1110,1100
 1100 N=N+1
      LVSD=1
      GO TO 1140
 1110 LVSD=-1
      NCT=NCT+1
      GO TO 1140
 1120 SL=-1.0
      LVSD=1
      GO TO 1140
 1130 NCT=NCT+1
 1140 COST=COS(THETA(I))
      SINT=SIN(THETA(I))
      TANA= TAN(ALPHAS(I))
      ABR=BETA*TANA
      IF(ABR-1.0)1150,2570,2570
 1150 IF(ABS(ABR)-EPS)1160,1160,1170
 1160 ABR=0.0
      SA=1.0
      GO TO 1180
 1170 SA=SIGN(1.,ABR)
 1180 APM2=ABR*ABR
      IN=I-NBODY
      NPT=NPART(I)
      TEST=ABS(Y(I,1,1))+ABS(ABS(THETA(I))-.5*PI)
C
C
C     INFLUENCED PANEL J
C
      DO 2430 J=1,NPANEL
      JN=J-NBODY
      U(J)=0.0
      V(J)=0.0
      W(J)=0.0
      VPM(J)=0.
      WPM(J)=0.
      VPMM(J)=0.
      WPMM(J)=0.
      SINA=SIN(ALPHAS(J))
      COSA=COS(ALPHAS(J))
C
      DO 2430M=1,NPT
      BBETAM(1)=(X(I,M,2)-X(I,M,1))/((Y(I,M,2)-Y(I,M,1))*COST
     1+(Z(I,M,2)-Z(I,M,1))*SINT)
      BBETAM(3)=(X(I,M,4)-X(I,M,3))/((Y(I,M,4)-Y(I,M,3))*COST
     1+(Z(I,M,4)-Z(I,M,3))*SINT)
      BPM1=BBETAM(1)
      BPM3=BBETAM(3)
      IF(BETA)1190,1200,1190
 1190 BBETAM(1)=BBETAM(1)/BETA
      BBETAM(3)=BBETAM(3)/BETA
 1200 BBETAM(2)=BBETAM(1)
      BBETAM(4)=BBETAM(3)
C
C     CORNER POINTS K OF INFLUENCING PANEL I
C
      DO 2420K=1,4
```

```
C
      R(K)=0.0
      C(K)=0.0
      Q(K)=0.0
      RPM(K)=0.
      CPM(K)=0.
      RPMM(K)=0.
      CPMM(K)=0.
      BPM=BBETAM(K)
      IF(LVSD)1240,1240,1210
 1210 IF(SL)1220,1240,1230
 1220 IF(K-2)2420,2420,1240
 1230 IF(K-2)1240,1240,2420
 1240 IF(ABS(BPM)-EPS)1260,1260,1250
 1250 SM=SIGN(1.0,BPM)
      GO TO 1270
 1260 SM=1.0
      BPM=0.0
 1270 IF (ABS(ABS(BPM)-1.0)-EPS)1280,1280,1290
 1280 BPM=SM
 1290 BPM=SM*BPM
      BPM2=BPM*BPM
      CONSTC=1.+APM2*BPM2
C
C     CONFIGURATION SYMMETRY CONDITION
C
      DO 2410ISIDE=1,ISYM
      IF(ISIDE.EQ.2.AND.TEST.LE.EPS) GO TO 2410
      SN=(-1.0)**ISIDE
      DELTAT=THETA(I)+THETA(J)*SN
      IF(ABS(DELTAT).LE.EPS) DELTAT=0.
      SINDT=SIN(DELTAT)
      COSDT=COS(DELTAT)
      DELTAY=SN*(-YC(J))-Y(I,M,K)
      DELTAZ=ZC(J)-Z(I,M,K)
      XPM=XCPT(J)-X(I,M,K)
      YPM=(DELTAY*COST+DELTAZ*SINT)*SM
      ZBR=DELTAZ*COST-DELTAY*SINT
      IF (ABS(ZBR)-EPS)1300,1300,1310
 1300 ZBR=0.0
 1310 SZ=SIGN(1.,ZBR)
      ZPM2=ZBR*ZBR
      IF (ABS(YPM)-EPS)1320,1320,1330
 1320 YPM=0.0
 1330 SY=SIGN(1.,YPM)
      XIPM=XPM/BETA
      XIPM2=XIPM*XIPM
      YPM2=YPM*YPM
      IF (THKW)1910,1340,1910
 1340 CONTINUE
C
C     VELOCITY COMPONENTS INDUCED BY PANELS PARALLEL TO X- AXIS
      ZPM=ZBR
      TERMAA=XIPM-BPM*YPM
      TERMA=TERMAA**2
      TERMB=YPM2+ZPM2
      TERMC=SQRT(TERMB)
      IF(XMACH-1.)1580,1810,1350
 1350 TERMD=BPM2-1.
      TERME=-TERMD
      TERMG=TERMA+TERMD*ZPM2
      IF(XIPM-TERMC)1360,1530,1370
 1360 IF(BPM2-1.)1510,1570,1570
 1370 IF(ZPM)1380,1410,1380
 1380 FTR1=(BPM*TERMB-XIPM*YPM)/SQRT(TERMB*TERMG)
```

```
      IF(FTR1+1.)1450,1450,1390
1390  IF(FTR1-1.)1400,1420,1420
1400  FTR1=ACOS(FTR1)*SZ
      GO TO 1460
1410  IF(YPM)1420,1440,1430
1420  FTR1=0.
      GO TO 1460
1430  IF(XIPM-BPM*YPM)1420,1440,1450
1440  FTR1=PI/2.
      GO TO 1460
1450  FTR1=PI*SZ
1460  IF(BPM2-1.)1470,1480,1480
1470  FTR2=-SQRT(TERME)*ACOS((BPM*XIPM-YPM)/SQRT(TERMG))
      GO TO 1490
1480  FTR2=SQRT(TERMD)*ACOSH((BPM*XIPM-YPM)/SQRT(TERMG))
1490  FTR4=SQRT(XIPM2-TERMB)/TERMB
      FTR5=ACOSH(XIPM/TERMC)
      D=CONSTB*(BPM*FTR5+YPM*FTR4-FTR2)
      S=CONSTB*(BPM*FTR1-ZPM*FTR4)
      P=FTR1/(4.*PI)
      GO TO 2370
1510  IF(YPM)1570,1570,1520
1520  IF (XIPM-(BPM*YPM+SQRT(TERME)*ABS(ZPM)))1570,1540,1530
1530  IF (YPM-BPM*XIPM)1570,1540,1550
1540  CT=.5
      GO TO 1560
1550  CT=1.0
1560  D=CONSTA*SQRT(TERME)*CT
      S=CONSTA*BPM*SZ*CT
      P=CT*SZ/4.
      GO TO 2370
1570  D=0.0
      S=0.0
      P=0.0
      GO TO 2370
1580  TERMD=1.+BPM2
      TERME=SQRT(TERMD)
      IF(ZPM)1680,1590,1680
1590  IF(TERMAA*SY)1600,1610,1620
1600  FTR1=0.
      GO TO 1630
1610  FTR1=PI/2.
      GO TO 1630
1620  FTR1=PI
1630  IF(YPM)1640,1720,1720
1640  TERMG=TERMAA/(YPM+BPM*XIPM)
      IF(ABS(TERMG)-.001)1650,1650,1720
1650  TERMH=TERMAA/ABS(YPM)
      IF(TERMH)1660,1670,1660
1660  FTR2=ALOG(TERMAA*TERMH/(2.*TERMD)*(1.+BPM*TERMH/TERMD))
      GO TO 1730
1670  FTR2=ALOG(TERMD)
      GO TO 1730
1680  FTR1=(BPM*TERMB-XIPM*YPM)/SQRT(TERMB*(TERMA+TERMD*ZPM2))
      IF(FTR1+1.)1740,1740,1690
1690  IF(FTR1-1.)1710,1700,1700
1700  FTR1=0.
      GO TO 1720
1710  FTR1=ACOS(FTR1)*SZ
1720  FTR2= YPM+BPM*XIPM+TERME*SQRT(XIPM2+TERMB)
      IF (FTR2 .LE. 0.0) GO TO 1650
      FTR2 = ALOG(FTR2)
1730  FTR3=(XIPM*SQRT(XIPM2+TERMB))/TERMB
      IF(FTR3)1760,1750,1760
1740  FTR1 = PI*SZ
```

```
      GO TO 1720
 1750 FTR4=0.
      GO TO 1770
 1760 FTR4=BPM*ALOG(FTR3)
 1770 IF(ZPM)1780,1790,1780
 1780 FTR5=ATAN(YPM/ZPM)
      GO TO 1800
 1790 FTR5=PI/2.*SY
 1800 P=(FTR1+FTR5)/(8.*PI)
      S=CONSTB*(BPM*(FTR1+FTR5)-ZPM*FTR3)/2.
      D=CONSTB*(YPM*FTR3+FTR4-TERME*FTR2)/2.
      GO TO 2370
 1810 IF(XPM)1570,1570,1820
 1820 TERMD=BPM**2
      TERMA=(XPM-BPM*YPM)**2
      IF(ZPM)1830,1840,1830
 1830 FTR1=ACOS((-XPM*YPM+BPM*TERMB)/SQRT(TERMB*(TERMA+TERMD*ZPM2)))*SZ
      GO TO 1880
 1840 IF(YPM)1850,1850,1860
 1850 FTR1=0.
      GO TO 1880
 1860 IF(XPM-BPM*YPM)1850,1850,1870
 1870 FTR1=PI
 1880 P=FTR1/(4.*PI)
      S=-(ABS(XPM)*ZPM/TERMB-BPM*FTR1)/(4.*PI)
      IF(BPM)1890,1900,1890
 1890 D=(ABS(XPM)*YPM/TERMB-BPM*ALOG(BPM*TERMC/SQRT(TERMA+ZPM2*TERMD)))
     1/(4.*PI)
      GO TO 2370
 1900 D=ABS(XPM)*YPM/(TERMB*PI*4.)
      GO TO 2370
 1910 CONTINUE
C
C     VELOCITY COMPONENTS INDUCED BY WING SOURCES
      ZPM=ZBR
      TERMAA=XIPM-BPM*YPM
      TERMA=(XIPM-BPM*YPM)**2
      TERMB=YPM2+ZPM2
      TERMC=SQRT(TERMB)
      IF(XMACH-1.)2190,2370,1920
 1920 TERMD=BPM2-1.
      TERME=-TERMD
      IF (XIPM-TERMC)1930,1930,1940
 1930 IF(TERMD)2100,1570,1570
 1940 FTR1=ACOSH(XIPM/TERMC)
      IF (TERMD)1950,1960,1970
 1950 FTR2=ACOS((BPM*XIPM-YPM)/SQRT(TERMA+TERMD*ZPM2))/SQRT(TERME)
      GO TO 1980
 1960 FTR2=SQRT(XIPM2-TERMB)/(XIPM-YPM)
      GO TO 1980
 1970 FTR2=ACOSH((BPM*XIPM-YPM)/SQRT(TERMA+TERMD*ZPM2))/SQRT(TERMD)
 1980 IF (ZPM)1990,2020,1990
 1990 FTR3=      (-XIPM*YPM+BPM*TERMB)/SQRT(TERMB*(TERMA+TERMD*ZPM2))
      IF (FTR3+1.0)2060,2060,2000
 2000 IF (FTR3-1.0)2010,2030,2030
 2010 FTR3=ACOS(FTR3)*SZ
      GO TO 2070
 2020 IF (YPM)2030,2030,2040
 2030 FTR3=0.
      GO TO 2070
 2040 IF (XIPM-BPM*YPM)2030,2050,2060
 2050 FTR3=PI/2.
      GO TO 2070
 2060 FTR3=PI*SZ
 2070 IF(LVSD)2080,2090,2090
```

```
2080 P=(TERMAA*FTR2+YPM*FTR1-ZPM*FTR3)/(CHORD(NCT)*PI)
     S=-CONSTD*(TERMAA*(FTR1-BPM*FTR2)+BPM*ZPM*FTR3
    1-SQRT(XIPM2-TERMB))/CHORD(NCT)
     D=-CONSTD*(ZPM*(BPM*FTR1-TERMD*FTR2)-TERMAA*FTR3)
    1/CHORD(NCT)
     GO TO 2370
2090 P=FTR2/(BETA*PI)
     S=-(FTR1-BPM*FTR2)/PI
     D=FTR3/PI
     GO TO 2370
2100 IF(YPM)1570,1570,2110
2110 IF (XIPM-TERMC)2120,2130,1940
2120 IF (XIPM-(BPM*YPM+SQRT(TERME)*ABS(ZPM)))1570,2140,2130
2130 IF(YPM-BPM*XIPM)1570,2140,2150
2140 CT=.5
     GO TO 2160
2150 CT=1.
2160 IF(LVSD)2170,2180,2180
2170 P=(TERMAA/SQRT(TERME)-ZPM*CT*SZ)/CHORD(NCT)
     S=BPM*P*BETA
     D=BETA*(TERMAA*CT*SZ-ZPM*SQRT(TERME))/CHORD(NCT)
     GO TO 2370
2180 P=CT/(SQRT(TERME)*BETA)
     S=BPM*P*BETA
     D=CT*SZ
     GO TO 2370
2190 TERMD=BPM2+1.
     TERME=SQRT(TERMD)
     FTR1=XIPM/TERMC
     IF(FTR1)2210,2200,2200
2200 FTR1=ASINH(FTR1)
     GO TO 2220
2210 FTR1=-ASINH(-FTR1)
2220 FTR2=(YPM+BPM*XIPM)/SQRT(TERMA+TERMD*ZPM2)
     IF(FTR2)2240,2230,2230
2230 FTR2=ASINH(FTR2)/TERME
     GO TO 2250
2240 FTR2=-ASINH(-FTR2)/TERME
2250 IF(ZPM)2260,2290,2260
2260 FTR3=(BPM*TERMB-XIPM*YPM)/SQRT(TERMB*(TERMA+TERMD*ZPM2))
     IF(FTR3+1.)2330,2330,2270
2270 IF(FTR3-1.)2280,2300,2300
2280 FTR3=ACOS(FTR3)*SZ
     GO TO 2340
2290 IF(YPM)2300,2300,2310
2300 FTR3=0.
     GO TO 2340
2310 IF(XIPM-BPM*YPM)2300,2320,2330
2320 FTR3=PI/2.
     GO TO 2340
2330 FTR3=PI*SZ
2340 FTR4=SQRT(XIPM2+TERMB)
     IF(LVSD)2350,2360,2360
2350 P=(TERMAA*FTR2+YPM*FTR1-ZPM*FTR3)/(CHORD(NCT)*2.*PI)
     S=-CONSTD*(TERMAA*(FTR1-BPM*FTR2)+BPM*ZPM*FTR3-FTR4)/
    1(2.*CHORD(NCT))
     D=-CONSTD*(ZPM*(BPM*FTR1-TERMD*FTR2)-TERMAA*FTR3)/CHORD(NCT)
     GO TO 2370
2360 P=FTR2/(2.*BETA*PI)
     S=-(FTR1-BPM*FTR2)/(2.*PI)
     D=FTR3/PI
2370 CONTINUE
     SIM=1.
     R(K)=R(K)+S*SINDT*SIM
     IF(THKW .NE. 1.0)  GO TO 2390
```


MCDONNELL DOUGLAS

```
      IF(DELTAT.EQ.0.0.AND.ZPM.EQ.0.0) D=0.0
      GO TO 2400
 2390 IF(I.EQ.J.AND.ISIDE.EQ.1) P=0.
      IF(ISIDE.GT.1) GO TO 2400
      IF(IN.LE.0)GO TO 2400
      VS2(IN)=0.
      IF(J.LT.I) GO TO 2400
      NA=(IN-1)/NR+1
      NB=(JN-1)/NR+1
      IF(NA.EQ.NB) S=0.
      IF(J.EQ.I) VS1(IN)=.25*BPM1
      IF(J.GT.I) VS2(IN)=.25*(BPM1-BPM3)
 2400 C(K)=C(K)+D*COSDT*SM*SIM
      Q(K)=Q(K)-P*SM*SIM
      RPM(K)=RPM(K)+S*SIM
      CPM(K)=CPM(K)+D*SM*SIM
      RPMM(K)=RPMM(K)-S*SN*SIM
      CPMM(K)=CPMM(K)-D*SM*SN*SIM
 2410 CONTINUE
 2420 CONTINUE
C
      U(J)=U(J)+Q(1)-Q(2)-Q(3)+Q(4)
      V(J)=V(J)+(R(1)-R(2)-R(3)+R(4))
      W(J)=W(J)+(C(1)-C(2)-C(3)+C(4))
      VPM(J)=VPM(J)+(RPM(1)-RPM(2)-RPM(3)+RPM(4))
      WPM(J)=WPM(J)+(CPM(1)-CPM(2)-CPM(3)+CPM(4))
      VPMM(J)=VPMM(J)+(RPMM(1)-RPMM(2)-RPMM(3)+RPMM(4))
      WPMM(J)=WPMM(J)+(CPMM(1)-CPMM(2)-CPMM(3)+CPMM(4))
 2430 CONTINUE
C
      IF(LVSD)2460,2440,2460
C
 2440 DO 2450 J=1,NPANEL
      UU(J)=U(J)
      VV(J)=VPMM(J)*COST-WPMM(J)*SINT
      WW(J)=WPM(J)*COST+VPM(J)*SINT
 2450 A(J)=V(J)+W(J)
C
      GO TO 2510
 2460 IF(IT)2490,2490,2470
C
 2470 DO 2480 J=1,NPANEL
      US(J)=U(J)
      VS(J)=VPMM(J)*COST-WPMM(J)*SINT
      WS(J)=WPM(J)*COST+VPM(J)*SINT
 2480 AS(J)=V(J)+W(J)
C
      IF(IT)2510,2510,1080
C
 2490 DO 2500 J=1,NPANEL
      UU(J)=US(J)-U(J)*SL
      VV(J)=VS(J)-(VPMM(J)*COST-WPMM(J)*SINT)*SL
      WW(J)=WS(J)-(WPM(J)*COST+VPM(J)*SINT)*SL
 2500 A(J)=AS(J)-(V(J)+W(J))*SL
C
      IF(LVSD)2470,2440,2510
 2510 WRITE(NTAPEA)(A(J),J=1,NPANEL)
      WRITE(NTAPEB)(UU(J),VV(J),WW(J),J=1,NPANEL)
      IF(IT)2530,2520,2530
 2520 IF(LVSD)1120,2440,2530
C
C *****************
 2530 CONTINUE
C *****************
C
```

```
      IF(KACE-2)2580,2580,2540                                          00004390
 2540 IF(NCT-NPANEL)2550,2580,2580                                      00004400
 2550 IF(THKW)2580,2560,2580                                            00004410
 2560 NC=NWING/NROW(2)                                                  00004420
      NR=NROW(2)                                                        00004430
      I1=NBODY+1                                                        00004440
      I2=NPANEL                                                         00004450
      NCT=NBODY                                                         00004460
      GO TO 1060                                                        00004470
C                                                                       00004480
 2570 WRITE(NTAPEO,2590)                                                00004490
 2580 CALL FOR EOF (NTAPEA)                                             00004500
      CALL FOR EOF (NTAPEB)                                             00004510
C                                                                       00004520
C -------------------                                                   00004530
      RETURN                                                            00004540
C -------------------                                                   00004550
 2590 FORMAT(1HO,41HERROR RETURN. WING LIES OUTSIDE MACH CONE)          00004560
      END                                                               -
CDECK INTAPE
      SUBROUTINE INTAPE                                                 00000010
C                                                                       00000020
C ..................................................................... 00000030
C     READS IN GEOMETRICAL DATA FROM PANELING AND DEFINITION LINKS      00000040
C ..................................................................... 00000050
C                                                                       00000060
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /DSXB/XBS1,XBS2,DSB1,DSB2                                  00000090
C                                                                       00000100
      COMMON /BLOCK/ALPHAS(210),AREA(210),A(210),ALPHAC(110),ALPHAT(110)00000110
     1,CHORD(210)                                                       00000120
     2,ISYM                                                             00000130
     3,NPART(210),NPANEL,NROW(2)                                        00000140
     4,THETA(210),TAIL,THKOMP                                           00000150
     5,U(210)                                                           00000160
     6,V(210),VPM(210),VV(210),VPMM(210)                                00000170
     7,W(210),WPM(210),WW(210),WPMM(210)                                00000180
     8,X(210,3,4),XBAR(210),XC(210)                                     00000190
     9,Y(210,3,4),YBAR(210),YC(210)                                     00000200
     A,Z(210,3,4),ZBAR(210),ZC(210)                                     00000210
      COMMON /WTHICK/THKW                                               00000220
      DIMENSION JSAVE(2),NSAVE(2),XB(51),R(51),ZDELTA(51)               00000230
      DIMENSION THETAB(11), THETB(11)
      DIMENSION XYZ(3)                                                  00000250
C                                                                       00000270
C                                                                       00000280
      READ (NTAPEI,1290) XMACH, SYM, THKOMP
C                                                                       00000330
      BCOND=0                                                           00000340
      KOND=BCOND                                                        00000350
      IF (SYM)1010,1000,1010                                            00000360
 1000 ISYM=1                                                            00000370
      GO TO 1020                                                        00000380
 1010 ISYM=2                                                            00000390
 1020 CONTINUE                                                          00000400
C                                                                       00000410
C     CODE TO INDICATE IF THE CASE INCLUDES A THICK WING                00000420
      THKW = THKOMP                                                     00000430
C                                                                       00000440
C     TEST FOR TYPE OF CONFIGURATION                                    00000450
      IF (NBODY)1040,1030,1040                                          00000460
C                                                                       00000470
C     WING ONLY CASE                                                    00000480
 1030 KACE=1                                                            00000490
```

```
      JSAVE(1)=0
      NSAVE(1)=NWING
      NPANEL=NWING
      NBODYS=0
      NRG=1
      GO TO 1070
C
 1040 CONTINUE
C
C     READ NUMBER OF BODY SOURCE SEGMENTS AND THE INDEX OF THE X -
C     COORDINATE AT LEADING EDGE OF THE WING
      READ (NTAPEB) (XYZ(I),I=1,3),ZA,S,T
      ALPHAA=XYZ(3)/XYZ(1)
      READ (NTAPEB) NBODYS,NXLE
C
C     READ BODY X-COORDINATES, BODY RADIUS, AND INCREMENT OF BODY CAMBER
      READ (NTAPEB) (XB(I),R(I),ZDELTA(I),I=1,NBODYS)
      REWIND NTAPEB
C
      IF (NBODY)1050,1280,1060
C
C     BODY ONLY CASE
 1050 KACE=2
      NPANEL=0
      NWING=0
      NWINGS = 0
      WRITE (NTAPEC) NBODY,NWING,XMACH,SYM,KACE,THKOMP
      GO TO 1160
C
C     WING AND BODY CASE
 1060 KACE=3
      JSAVE(1)=0
      NSAVE(1)=NBODY
      JSAVE(2)=NBODY
      NSAVE(2)=NWING
      NPANEL=NWING+NBODY
      NRG=2
C
 1070 CONTINUE
C
C
      DO 1120 L=1,NRG
      J=JSAVE(L)
      N=NSAVE(L)
C
      DO 1100I=1,N
      J=J+1
C
C     READS IN PANEL COORDINATES
      READ (NTAPEC) NPN,NPART(J),(X(J,1,K),Y(J,1,K),Z(J,1,K),K=1,4)
      IF (NPART(J)-1)1080,1100,1080
 1080 NPT=NPART(J)
      DO 1090M=2,NPT
      READ (NTAPEC)(X(J,M,K),Y(J,M,K),Z(J,M,K),K=1,4)
 1090 CONTINUE
 1100 CONTINUE
C
      J=JSAVE(L)
      DO 1110I=1,N
      J=J+1
C
C     READS IN COORDINATES OF CENTROIDS, CONTROL POINTS, AREAS, THETAS,
C     ALPHAS, AND CHORD LENGTHS
      READ (NTAPEC) NPN,XBAR(J),YBAR(J),ZBAR(J),XC(J),YC(J),ZC(J),AREA(J
     1),THETA(J),ALPHAS(J),CHORD(J)
```

```
 1110 CONTINUE                                                          00001150
C                                                                       00001160
C     READS IN THE NUMBER OF PANELS IN A COLUMN AND THE LOCATION OF THE 00001170
C     CONTROL POINT                                                     00001180
      READ (NTAPEC) NROW(L),XCTP                                        00001190
 1120 CONTINUE                                                          00001200
      IF (NBODY .NE. 0) READ (NTAPEC) NTHETB                            00001210
      IF (NBODY .NE. 0) READ (NTAPEC) (THETB(N), N = 1, NTHETB)         00001220
      REWIND NTAPEC                                                     00001230
C                                                                       00001240
      DO 1140 J=1,NWING                                                 00001250
      JJ=J+NBODY                                                        00001260
C                                                                       00001270
C     STORES THICKNESS SLOPES AND CAMBER SLOPES                         00001280
      ALPHAT(J)=0.0                                                     00001290
 1140 ALPHAC(J)=0.0                                                     00001300
      NS=NBODY+1                                                        00001310
C     WRITE 1ST FILE ON TAPEC                                           00001360
      WRITE (NTAPEC) NBODY,NWING,XMACH,SYM,KACE,THKOMP                  00001370
C                                                                       00001380
C                                                                       00001390
C     WRITES COORDINATES OF CENTROIDS, CONTROL POINTS, AREAS, THETAS,   00001400
C     ALPHAS, CHORD LENGTHS, NUMBER OF REGIONS, NUMBER OF ROWS PER      00001410
C     COLUMN IN A REGION, AND CONTROL POINT LOCATION                    00001420
      WRITE (NTAPEC) (I,XBAR(I),YBAR(I),ZBAR(I),XC(I),YC(I),ZC(I),AREA(I00001430
     1),THETA(I),ALPHAS(I),CHORD(I),I=1,NPANEL),NRG,(NROW(I),I=1,NRG),XC00001440
     2TP                                                                00001450
      CALL FOR EOF (NTAPEC)                                             00001460
C                                                                       00001470
C     WRITE THICKNESS SLOPES. 2ND FILE ON TAPEC                         00001480
      WRITE (NTAPEC) (ALPHAT(I),I=1,NWING)                              00001490
      CALL FOR EOF (NTAPEC)                                             00001500
C                                                                       00001510
C     WRITE CAMBER SLOPES. 3RD FILE ON TAPEC                            00001520
      WRITE (NTAPEC) (ALPHAC(I),I=1,NWING)                              00001530
      CALL FOR EOF (NTAPEC)                                             00001540
C                                                                       00001550
      IF (NBODY)1160,1230,1160                                          00001560
 1160 CONTINUE                                                          00001570
C                                                                       00001580
C     WRITE SLOPE OF BODY AXIS WITH RESPECT TO THE DEFINING AXIS        00001590
C     FIRST PART OF 4TH FILE ON TAPEC                                   00001600
      WRITE (NTAPEC) ALPHAA,DSB1,DSB2,XBS1,XBS2                         00001610
C                                                                       00001620
C     WRITE NUMBER OF BODY SOURCE SEGMENTS, INDEX OF X- COORDINATE AT   00001630
C     LEADING EDGE OF WING, HEIGHT OF WING PLANE ABOVE BODY AXIS, X     00001640
C     COORDINATE OF BODY.  SECOND PART OF 4TH FILE ON TAPEC.            00001650
      WRITE (NTAPEC) NBODYS,NXLE,ZA,(XB(I),I=1,NBODYS)                  00001660
      IF (NWING)1180,1170,1180                                          00001670
 1170 CONTINUE                                                          00001680
C                                                                       00001690
C     FOR BODY ONLY CASE, READ NUMBER OF THETAS REPRESENTING BODY       00001700
      READ (NTAPED) NDUMMY,NDUMMY,NDUMMY,NTHETB                         00001710
C                                                                       00001720
C     READ BODY THETAS                                                  00001730
      READ (NTAPED) (THETB(J),J=1,NTHETB)                               00001740
      REWIND NTAPED                                                     00001750
 1180 DO 1190 J = 1,NTHETB                                              00001760
 1190 THETB(J) = 57.29578*THETB(J)                                      00001770
      IF (NWING .EQ. 0) GO TO 1210                                      00001780
C     FOR WING-BODY CASE COMPUTE THETAS FROM PANEL REPRESENTATION OF BOD00001790
      NROWB = NROW(1)                                                   00001800
      NTHETA = NBODY/NROWB                                              00001810
      DO 1200 J = 1, NTHETA                                             00001820
      JJ = (J-1)*NROWB+1                                                00001830
```

```
 1200 THETAB(J) = ABS(THETA(JJ))*57.29578                                00001840
C                                                                        00001850
C     WRITE NUMBER OF THETAS, AND THETAS. 3RD PART OF 4TH FILE ON C      00001860
      WRITE (NTAPEC) NTHETA,(THETAB(I),I=1,NTHETA)                       00001870
 1210 WRITE (NTAPEC) NTHETB, (THETB(I),I=1,NTHETB)                       00001880
C                                                                        00001890
C     WRITE RADIUS AND CAMBER INCREMENT OF BODY. LAST PART OF 4TH FILE   00001900
      WRITE (NTAPEC) (R(I),I=1,NBODYS)                                   00001910
      WRITE (NTAPEC) (ZDELTA(I),I=1,NBODYS)                              00001920
      CALL FOR EOF (NTAPEC)                                              00001940
      IF (NBODY)1220,1220,1240                                           00001950
 1220 CONTINUE                                                           00001960
      CALL FOR EOF (NTAPEC)                                              00001970
      CALL FOR EOF (NTAPEC)                                              00001980
 1230 CONTINUE                                                           00001990
      CALL FOR EOF (NTAPEC)                                              00002000
C                                                                        00002010
C     STORE GEOMETRY DATA ON SCRATCH/SAVE TAPE                           00002020
C     FOR USE IN FLOW VISUALIZATION LINK                                 00002030
 1240 CONTINUE                                                           00002040
      IF (NBODY .NE.-1) NWINGS=NWING+NWING/NROW(NRG)                     00002050
C     WRITE 5TH FILE ON TAPEC                                            00002060
      WRITE (NTAPEC) KACE,NPANEL,NBODY,NWING,NBODYS,NWINGS               00002070
     1,NROW,XMACH,SYM                                                    00002080
      CALL FOR EOF (NTAPEC)                                              00002090
C                                                                        00002100
      IF (NPANEL)1270,1270,1250                                          00002110
 1250 CONTINUE                                                           00002120
      WRITE (NTAPEC) NPANEL                                              00002130
      DO 1260 J=1,NPANEL                                                 00002140
      WRITE (NTAPEC) NPART(J)                                            00002150
      NPT=NPART(J)                                                       00002160
      WRITE (NTAPEC) ((X(J,M,K),Y(J,M,K),Z(J,M,K),K=1,4),M=1,NPT)        00002170
 1260 CONTINUE                                                           00002180
      WRITE (NTAPEC) (XBAR(J),YBAR(J),ZBAR(J),J=1,NPANEL)                00002190
      WRITE (NTAPEC) (XC(J),YC(J),ZC(J),J=1,NPANEL)                      00002200
      WRITE (NTAPEC) (ALPHAS(J),THETA(J),CHORD(J),J=1,NPANEL)            00002210
 1270 CONTINUE                                                           00002220
      CALL FOR EOF (NTAPEC)                                              00002230
 1280 CONTINUE                                                           00002240
C                                                                        00002250
C                                                                        00002260
C                                                                        00002270
C ------------------                                                     00002280
      RETURN                                                             00002290
C ------------------                                                     00002300
 1290 FORMAT(7F10.0)                                                     00002310
      END                                                                -
CDECK REDUCE
      OVERLAY(WOOD,4,0)
      PROGRAM REDUCE
C                                                                        REDU1010
C ...................................................................... REDU1020
C     REDUCE AERODYNAMIC INFLUENCE COEFFICIENTS MATRIX AND FORM THE      REDU1030
C     (D) AND (E) MATRICES                                               REDU1040
C ...................................................................... REDU1050
C                                                                        REDU1060
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                        REDU1090
      DIMENSION AB(100),ABB(100,100),ABW(100,25),AWB(110,100),AWW(110,25REDU1100
     1),E(100,25),D(110),NSIZE(5)                                        REDU1110
C                                                                        REDU1120
      EQUIVALENCE (ABB(1,1),AWB(1,1))                                    00000140
C                                                                        REDU1140
```

```
C
C       COMPUTE SIZE OF PARTIONS -
        MAXP=25
        CALL SIZE (NWING,MAXP,NPART,NSIZE)
        DO 1000 J=1,NBODY
 1000   READ (NTAPED) (AWB(I,J),I=1,NWING)
        CALL FSF(1,NTAPED,IRR)
C
C       FORM (D) MATRIX
        DO 1020 K=1,NBODY
        READ (NTAPED) (AB(I),I=1,NBODY)
        DO 1010 I=1,NWING
        D(I)=0.0
        DO 1010 J=1,NBODY
        D(I)=D(I)+AWB(I,J)*AB(J)
 1010   CONTINUE
        WRITE (NTAPEF) (D(I),I=1,NWING)
 1020   CONTINUE
        CALL FOR EOF (NTAPEF)
        REWIND NTAPED
        CALL FSF(1,NTAPED,IRR)
        DO 1030 J=1,NBODY
 1030   READ (NTAPED) (ABB(I,J),I=1,NBODY)
C
C       COPY INVERSE BODY MATRIX
        DO 1040 J=1,NBODY
 1040   WRITE (NTAPEF) (ABB(I,J),I=1,NBODY)
C
C       MATRIX REDUCTION
        DO 1130 L=1,NPART
        NS=NSIZE(L)
        IF (L .EQ. 1) GO TO 1060
        DO 1050 J=1,NBODY
 1050   READ (NTAPED) (ABB(I,J),I=1,NBODY)
 1060   REWIND NTAPED
        DO 1070 K=1,NS
 1070   READ (NTAPEA)(ABW(I,K),I=1,NBODY),(AWW(I,K),I=1,NWING)
C
C       FORM (E) MATRIX
        DO 1080 K=1,NS
        DO 1080 M=1,NBODY
        E(M,K)=0.
        DO 1080 N=1,NBODY
        E(M,K)=E(M,K)-ABB(M,N)*ABW(N,K)
 1080   CONTINUE
        DO 1090 J=1,NBODY
 1090   READ (NTAPED)(AWB(I,J),I=1,NWING)
        CALL FSF(1,NTAPED,IRR)
        DO 1100 K=1,NS
        DO 1100 I=1,NWING
        DO 1100 M=1,NBODY
        AWW(I,K)=AWW(I,K)+AWB(I,M)*E(M,K)
 1100   CONTINUE
C
        DO 1110 K=1,NS
 1110   WRITE (NTAPEF)(E(I,K),I=1,NBODY)
        DO 1120 K=1,NS
 1120   WRITE (NTAPEE)(AWW(I,K),I=1,NWING)
 1130   CONTINUE
        REWIND NTAPEA
        REWIND NTAPED
        REWIND NTAPEE
        CALL FOR EOF (NTAPEF)
C
        REWIND NTAPEF
```

```
C                                                                       REDU1800
C -----------------                                                     REDU1810
C -----------------                                                     REDU1830
      END                                                                      -
CDECK SIZE
      SUBROUTINE SIZE(MM,NMAX,NPART,NSIZE)                              SIZE1000
C                                                                       SIZE1010
C ..................................................................... SIZE1020
C                                                                       SIZE1030
C ..................................................................... SIZE1040
C                                                                       SIZE1050
      DIMENSION NSIZE(1)                                                SIZE1060
C                                                                       SIZE1070
C     MM     SIZE OF MATRIX                                             SIZE1080
C     NMAX   MAXIMUM SIZE OF PARTITIONS                                 SIZE1090
C     NPART  NUMBER OF PARTITIONS                                       SIZE1100
C     NSIZE  ARRAY OF THE SIZE OF EACH PARTITION                        SIZE1110
C                                                                       SIZE1120
C                                                                       SIZE1130
      NPMIN=2                                                           SIZE1140
      NT=MM                                                             SIZE1150
      NPART=MAX0 (NPMIN,1+(NT-1)/NMAX)                                  SIZE1160
      N=NPART                                                           SIZE1170
 1000 NSIZE(N)=NT/N                                                     SIZE1180
      NT=NT-NSIZE(N)                                                    SIZE1190
      N=N-1                                                             SIZE1200
      IF (N)1010,1010,1000                                              SIZE1210
C                                                                       SIZE1220
C -----------------                                                     SIZE1230
 1010 RETURN                                                            SIZE1240
C -----------------                                                     SIZE1250
      END                                                                      -
CDECK PARTV
      OVERLAY(WOOD,5,0)
      PROGRAM PARTV
C                                                                       PART1010
C ..................................................................... PART1020
C     PARTITION VELOCITY COMPONENTS MATRIX                              PART1030
C ..................................................................... PART1040
C                                                                       PART1050
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                       PART1080
      DIMENSION U(110),V(110),W(110),UU(110),VV(110),WW(110)            PART1090
C                                                                       PART1100
C                                                                       PART1110
      CALL FSF(1,NTAPEA,IRR)                                            PART1120
      CALL FSF(1,NTAPEB,IRR)                                            PART1130
      DO 1000 J=1,NBODY                                                 PART1140
      READ (NTAPEB) (U(I),V(I),W(I),I=1,NBODY),(UU(I),VV(I),WW(I),I=1,  PART1150
     1NWING)                                                            PART1160
      WRITE (NTAPED) (U(I),V(I),W(I),I=1,NBODY)                         PART1170
      WRITE (NTAPEA) (UU(I),VV(I),WW(I),I=1,NWING)                      PART1180
 1000 CONTINUE                                                          PART1190
      CALL FOR EOF (NTAPED)                                             PART1200
      REWIND NTAPEA                                                     PART1210
      CALL FSF(1,NTAPEA,IRR)                                            PART1220
C                                                                       PART1230
      DO 1010 J=1,NBODY                                                 PART1240
      READ (NTAPEA) (UU(I),VV(I),WW(I),I=1,NWING)                       PART1250
      WRITE (NTAPED) (UU(I),VV(I),WW(I),I=1,NWING)                      PART1260
 1010 CONTINUE                                                          PART1270
C                                                                       PART1280
      CALL FOR EOF (NTAPED)                                             PART1290
      REWIND NTAPEA                                                     PART1300
```

```
      CALL FSF(1,NTAPEA,IRR)                                            PART1310
C                                                                       PART1320
      DO 1020 J=1,NWING                                                 PART1330
      READ (NTAPEB) (U(I),V(I),W(I),I=1,NBODY),(UU(I),VV(I),WW(I),I=1,NWPART1340
     1ING)                                                              PART1350
      WRITE (NTAPED) (U(I),V(I),W(I),I=1,NBODY)                         PART1360
      WRITE (NTAPEA) (UU(I),VV(I),WW(I),I=1,NWING)                      PART1370
 1020 CONTINUE                                                          PART1380
C                                                                       PART1390
      REWIND NTAPEA                                                     PART1400
      REWIND NTAPEB                                                     PART1410
      CALL FOR EOF (NTAPED)                                             PART1420
      CALL FSF(1,NTAPEA,IRR)                                            PART1430
C                                                                       PART1440
      DO 1030 J=1,NWING                                                 PART1450
      READ (NTAPEA) (UU(I),VV(I),WW(I),I=1,NWING)                       PART1460
      WRITE (NTAPED) (UU(I),VV(I),WW(I),I=1,NWING)                      PART1470
 1030 CONTINUE                                                          PART1480
      CALL FOR EOF (NTAPED)                                             PART1490
      REWIND NTAPEA                                                     PART1500
      REWIND NTAPED                                                     PART1510
C                                                                       PART1520
C ------------------                                                    PART1530
C ------------------                                                    PART1550
      END                                                                      -
CDECK WOOD60
      OVERLAY(WOOD,6,0)
      PROGRAM WOOD60
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1           NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /TRNSFR/ IOVR10,IOVR60,IOVR70
      COMMON /COM60/ ISAVET
      IF (IOVR60.EQ.2) GO TO 100
      CALL SAVTAP(ISAVET)
      GO TO 999
  100 CALL USETAP
 999  CONTINUE
      END
CDECK SAVTAP
      SUBROUTINE SAVTAP(ISAVET)                                         SAVT1000
C                                                                       SAVT1010
C ..................................................................... SAVT1020
C     WRITES AERODYNAMIC MATRICES ON A LOGICAL TAPE FOR FUTURE USE      SAVT1030
C ..................................................................... SAVT1040
C                                                                       SAVT1050
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1           NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /WTHICK/THKW                                               SAVT1080
      DIMENSION A(210),B(210),C(210)                                    SAVT1090
      DIMENSION DUM(10),NROW(2)                                         SAVT1100
C                                                                       SAVT1110
      IF (ISAVET)1010,1010,1000                                         SAVT1120
 1000 READ (NTAPEC)NBODY,NWING,XMACH,SYM,KACE,THKW                      SAVT1130
C                                                                       SAVT1140
C *****************                                                     SAVT1150
      GO TO (1020,1010,1020),KACE                                       SAVT1160
C                                                                       SAVT1170
 1010 CALL FSF(6,NTAPEC,IRR)                                            SAVT1180
      GO TO 1090                                                        SAVT1190
 1020 NPANEL=NBODY+NWING                                                SAVT1200
      READ (NTAPEC) ( M, ( DUM(J), J=1,10), I=1,NPANEL), NRG,
     1(NROW(I),I=1,NRG),DUMMY                                           SAVT1220
      NCOLW=NWING/NROW(NRG)                                             SAVT1230
      CALL FSF( 6,NTAPEC,IRR)                                           SAVT1240
      GO TO (1030,1090,1060),KACE                                       SAVT1250
```

```
 1030 IF (THKW .EQ. 0.) GO TO 1040                                       SAVT1260
      CALL TTAPE (0,NWING,NWING+NCOLW,NTAPEA,NTAPEC,A,B,C)               SAVT1270
 1040 CALL FSF(1,NTAPEA,IRR)                                             SAVT1280
      CALL TTAPE(0,NWING,NWING,NTAPEA,NTAPEC,A,B,C)                      SAVT1290
      CALL FSF(1,NTAPEA,IRR)                                             SAVT1300
      CALL TTAPE(0,NWING+1,NWING+1,NTAPEA,NTAPEC,A,B,C)                  SAVT1310
      CALL FSF(1,NTAPEA,IRR)                                             SAVT1320
      CALL TTAPE(0,NWING+2,NWING+2,NTAPEA,NTAPEC,A,B,C)                  SAVT1330
      REWIND NTAPEA                                                      SAVT1340
      IF (THKW .EQ. 0.) GO TO 1050                                       SAVT1350
      CALL TTAPE(1,NWING,NWING+NCOLW,NTAPEB,NTAPEC,A,B,C)                SAVT1360
 1050 CALL FSF(1,NTAPEB,IRR)                                             SAVT1370
      CALL TTAPE(1,NWING,NWING,NTAPEB,NTAPEC,A,B,C)                      SAVT1380
      REWIND NTAPEB                                                      SAVT1390
      CALL TTAPE(0,NWING,NWING,NTAPEE,NTAPEC,A,B,C)                      SAVT1400
      CALL FSF(1,NTAPEE,IRR)                                             SAVT1410
      CALL TTAPE(0,NWING,NWING,NTAPEE,NTAPEC,A,B,C)                      SAVT1420
      REWIND NTAPEE                                                      SAVT1430
      GO TO 1090                                                         SAVT1440
C                                                                        SAVT1450
 1060 IF (THKW .EQ. 0.) GO TO 1070                                       SAVT1460
      CALL TTAPE(0,NPANEL,NWING +NCOLW,NTAPEA,NTAPEC,A,B,C)              SAVT1470
 1070 CALL FSF(1,NTAPEA,IRR)                                             SAVT1480
      CALL TTAPE(0,NWING,NWING,NTAPEA,NTAPEC,A,B,C)                      SAVT1490
      CALL FSF(1,NTAPEA,IRR)                                             SAVT1500
      CALL TTAPE(0,NWING+1,NWING+1,NTAPEA,NTAPEC,A,B,C)                  SAVT1510
      CALL FSF(1,NTAPEA,IRR)                                             SAVT1520
      CALL TTAPE(0,NWING+2,NWING+2,NTAPEA,NTAPEC,A,B,C)                  SAVT1530
      REWIND NTAPEA                                                      SAVT1540
      IF (THKW .EQ. 0.) GO TO 1080                                       SAVT1550
      CALL TTAPE(1,NPANEL,NWING +NCOLW,NTAPEB,NTAPEC,A,B,C)              SAVT1560
 1080 REWIND NTAPEB                                                      SAVT1570
      CALL TTAPE(1,NBODY,NBODY,NTAPED,NTAPEC,A,B,C)                      SAVT1580
      CALL FSF(1,NTAPED,IRR)                                             SAVT1590
      CALL TTAPE(1,NWING,NBODY,NTAPED,NTAPEC,A,B,C)                      SAVT1600
      CALL FSF(1,NTAPED,IRR)                                             SAVT1610
      CALL TTAPE(1,NBODY,NWING,NTAPED,NTAPEC,A,B,C)                      SAVT1620
      CALL FSF(1,NTAPED,IRR)                                             SAVT1630
      CALL TTAPE(1,NWING,NWING,NTAPED,NTAPEC,A,B,C)                      SAVT1640
      REWIND NTAPED                                                      SAVT1650
      CALL TTAPE(0,NWING,NWING,NTAPEE,NTAPEC,A,B,C)                      SAVT1660
      CALL FSF(1,NTAPEE,IRR)                                             SAVT1670
      CALL TTAPE(0,NWING,NWING,NTAPEE,NTAPEC,A,B,C)                      SAVT1680
      REWIND NTAPEE                                                      SAVT1690
      CALL TTAPE(0,NWING,NBODY,NTAPEF,NTAPEC,A,B,C)                      SAVT1700
      CALL FSF(1,NTAPEF,IRR)                                             SAVT1710
      CALL TTAPE(0,NBODY,NBODY,NTAPEF,NTAPEC,A,B,C)                      SAVT1720
      CALL TTAPE(0,NBODY,NWING,NTAPEF,NTAPEC,A,B,C)                      SAVT1730
      REWIND NTAPEF                                                      SAVT1740
C                                                                        SAVT1750
 1090 CONTINUE                                                           SAVT1760
      CALL FOR EOF (NTAPEC)                                              SAVT1770
      REWIND NTAPEC                                                      SAVT1780
C                                                                        SAVT1790
C -----------------                                                      SAVT1800
      RETURN                                                             SAVT1810
C -----------------                                                      SAVT1820
      END                                                                -
CDECK USETAP
      SUBROUTINE USETAP                                                  USET1000
C                                                                        USET1010
C ...................................................................... USET1020
C      READS AERODYNAMIC MATRICES FROM SAVE TAPE AND REWRITES            USET1030
C ...................................................................... USET1040
C                                                                        USET1050
```

```
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1         NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /WTHICK/THKW                                               USET1080
      DIMENSION A(210),B(210),C(210)                                    USET1090
      DIMENSION DUM(10),NROW(2)                                         USET1100
C                                                                       USET1110
C                                                                       USET1120
      READ(NTAPEC)NBODY,NWING,XMACH,SYM,KACE,THKW                       USET1130
C                                                                       USET1140
      GO TO (1000,1070,1000),KACE                                       USET1150
 1000 NPANEL=NBODY+NWING                                                USET1160
      READ (NTAPEC) ( M, ( DUM(J), J=1,10), I=1,NPANEL), NRG,
     1(NROW(I),I=1,NRG),DUMMY                                           USET1180
      NCOLW=NWING/NROW(NRG)                                             USET1190
      CALL FSF( 6,NTAPEC,IRR)                                           USET1200
C                                                                       USET1210
      GO TO (1010,1070,1040),KACE                                       USET1220
 1010 IF (THKW .EQ. 0.) GO TO 1020                                      USET1230
      CALL TTAPE(0,NWING,NWING+NCOLW,NTAPEC,NTAPEA,A,B,C)               USET1240
 1020 CALL FOR EOF (NTAPEA)                                             USET1250
      CALL TTAPE(0,NWING,NWING,NTAPEC,NTAPEA,A,B,C)                     USET1260
      CALL FOR EOF (NTAPEA)                                             USET1270
      CALL TTAPE(0,NWING+1,NWING+1,NTAPEC,NTAPEA,A,B,C)                 USET1280
      CALL FOR EOF (NTAPEA)                                             USET1290
      CALL TTAPE(0,NWING+2,NWING+2,NTAPEC,NTAPEA,A,B,C)                 USET1300
      CALL FOR EOF (NTAPEA)                                             USET1310
      REWIND NTAPEA                                                     USET1320
      IF (THKW .EQ. 0.) GO TO 1030                                      USET1330
      CALL TTAPE(1,NWING,NWING+NCOLW,NTAPEC,NTAPEB,A,B,C)               USET1340
 1030 CALL FOR EOF (NTAPEB)                                             USET1350
      CALL TTAPE(1,NWING,NWING,NTAPEC,NTAPEB,A,B,C)                     USET1360
      CALL FOR EOF (NTAPEB)                                             USET1370
      REWIND NTAPEB                                                     USET1380
      CALL TTAPE(0,NWING,NWING,NTAPEC,NTAPEE,A,B,C)                     USET1390
      CALL FOR EOF (NTAPEE)                                             USET1400
      CALL TTAPE(0,NWING,NWING,NTAPEC,NTAPEE,A,B,C)                     USET1410
      CALL FOR EOF (NTAPEE)                                             USET1420
      REWIND NTAPEE                                                     USET1430
      GO TO 1070                                                        USET1440
 1040 IF (THKW .EQ. 0.) GO TO 1050                                      USET1450
      CALL TTAPE(0,NPANEL,NWING +NCOLW,NTAPEC,NTAPEA,A,B,C)             USET1460
 1050 CALL FOR EOF (NTAPEA)                                             USET1470
      CALL TTAPE(0,NWING,NWING,NTAPEC,NTAPEA,A,B,C)                     USET1480
      CALL FOR EOF (NTAPEA)                                             USET1490
      CALL TTAPE(0,NWING+1,NWING+1,NTAPEC,NTAPEA,A,B,C)                 USET1500
      CALL FOR EOF (NTAPEA)                                             USET1510
      CALL TTAPE(0,NWING+2,NWING+2,NTAPEC,NTAPEA,A,B,C)                 USET1520
      CALL FOR EOF (NTAPEA)                                             USET1530
      REWIND NTAPEA                                                     USET1540
      IF (THKW .EQ. 0.) GO TO 1060                                      USET1550
      CALL TTAPE(1,NPANEL,NWING +NCOLW,NTAPEC,NTAPEB,A,B,C)             USET1560
 1060 CALL FOR EOF (NTAPEB)                                             USET1570
      REWIND NTAPEB                                                     USET1580
      CALL TTAPE(1,NBODY,NBODY,NTAPEC,NTAPED,A,B,C)                     USET1590
      CALL FOR EOF (NTAPED)                                             USET1600
      CALL TTAPE(1,NWING,NBODY,NTAPEC,NTAPED,A,B,C)                     USET1610
      CALL FOR EOF (NTAPED)                                             USET1620
      CALL TTAPE(1,NBODY,NWING,NTAPEC,NTAPED,A,B,C)                     USET1630
      CALL FOR EOF (NTAPED)                                             USET1640
      CALL TTAPE(1,NWING,NWING,NTAPEC,NTAPED,A,B,C)                     USET1650
      CALL FOR EOF (NTAPED)                                             USET1660
      REWIND NTAPED                                                     USET1670
      CALL TTAPE(0,NWING,NWING,NTAPEC,NTAPEE,A,B,C)                     USET1680
      CALL FOR EOF (NTAPEE)                                             USET1690
      CALL TTAPE(0,NWING,NWING,NTAPEC,NTAPEE,A,B,C)                     USET1700
```

```
      CALL FOR EOF (NTAPEE)                                             USET1710
      REWIND NTAPEE                                                     USET1720
      CALL TTAPE(0,NWING,NBODY,NTAPEC,NTAPEF,A,B,C)                     USET1730
      CALL FOR EOF (NTAPEF)                                             USET1740
      CALL TTAPE(0,NBODY,NBODY,NTAPEC,NTAPEF,A,B,C)                     USET1750
      CALL TTAPE(0,NBODY,NWING,NTAPEC,NTAPEF,A,B,C)                     USET1760
      CALL FOR EOF (NTAPEF)                                             USET1770
      REWIND NTAPEF                                                     USET1780
 1070 CONTINUE                                                          USET1790
      REWIND NTAPEC                                                     USET1800
C                                                                       USET1810
C ------------------                                                    USET1820
      RETURN                                                            USET1830
C ------------------                                                    USET1840
      END                                                                      -
CDECK TTAPE
      SUBROUTINE TTAPE(KIND,IN,JN,NTIN,NTOUT,A,B,C)                     TTAP1000
C                                                                       TTAP1010
C ..................................................................... TTAP1020
C     COPIES MATRICES FROM TAPE TO TAPE                                 TTAP1030
C ..................................................................... TTAP1040
C                                                                       TTAP1050
      DIMENSION A(1),B(1),C(1)                                          TTAP1060
C                                                                       TTAP1070
      IF(KIND)1020,1000,1020                                            TTAP1080
 1000 CONTINUE                                                          TTAP1090
      DO 1010 J=1,JN                                                    TTAP1100
      READ(NTIN)(A(I),I=1,IN)                                           TTAP1110
 1010 WRITE(NTOUT)(A(I),I=1,IN)                                         TTAP1120
C                                                                       TTAP1130
      GO TO 1040                                                        TTAP1140
 1020 CONTINUE                                                          TTAP1150
      DO 1030 J=1,JN                                                    TTAP1160
      READ(NTIN)(A(I),B(I),C(I),I=1,IN)                                 TTAP1170
 1030 WRITE(NTOUT)(A(I),B(I),C(I),I=1,IN)                               TTAP1180
C                                                                       TTAP1190
 1040 CONTINUE                                                          TTAP1200
C                                                                       TTAP1210
C ------------------                                                    TTAP1220
      RETURN                                                            TTAP1230
C ------------------                                                    TTAP1240
      END                                                                      -
CDECK WOOD70
      OVERLAY(WANG, 7, 0)
      PROGRAM WANG70
C
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1        NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON /AVAR/  A(210),ACB(21),ABX(100),AWX(110),AREA(210),ARN(2),A
     1               RWT(20),ALPHA(210),ALPHAB(210),ALPHAS(210),ALPHAT(1
     2               10),ALPHAW(110),ALPHAX(110),AWS(110),ABT(100),ANIN(
     3               210,2),ALPHAA,ALPHAD,ARA,ARADEG,ARB,ARW,ARAS,AT,AAA
      COMMON /BODS/ NUMBOD,XNOS(3),YNOS(3),ZNOS(3),NESTAT(3)
      COMMON /EVAR/  B(210) , BBCL, BBCD, BBCM
      COMMON /CVAR/  C(210),CHORD(210),CL(210),CPNN(51,11,2),CLS(210)
     1               ,CDR,CASE,CPCALC,CBAR,CONSNT,CLBAR,CLX,CLM,CDM
      COMMON /DVAR/  D(210),DZDXB(51), DADEG, DARAD
      COMMON /IVAR/  IPOLAR
      COMMON /KVAR/  KASE,KONFIG,KPOLAR
      COMMON /NVAR/  NFMT(9),NROW(2)
     1               ,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,NTHETS,
     2                   NXLE,NRG,NPOLR ,NCLX
      COMMON /PVAR/  POLAR
```

```
      COMMON /RVAR/  R(51),RN(51,2), RATIOX, RFAREA
      COMMON /SVAR/  SEMIS,SLC
      COMMON /TVAR/  TITLE(20),THETA(210),THETAB(11),THETAS(11),THETAN(
     1          11,2),TCL(11),TCD(11),TCM(11),THICK,TWIST
      COMMON /UVAR/  UBWT(100),UWBT(110),UWWT(110),UNCL(210,2)
      COMMON /VVAR/  VBWT(100),VWBT(110),VWWT(110),VNCL(210,2)
     1               ,VOUT
      COMMON /WVAR/  WBWT(100),WWBT(110),WWWT(110),WNCL(210,2)
      COMMON /XVAR/  XBAR(210),XC(210),XNI(2),XNXN(2),XNTN(2),XB(51),XN(
     1               51,2),XNNCD(2),XNNCL(2),XNNCM(2),XYZ(3),XCL(11),XCD
     2               (11), XNACEL, XP, XCPBAR
      COMMON /YVAR/  YBAR(210),YC(210),YNI(2)
      COMMON /ZVAR/  ZBAR(210),ZC(210),ZNI(2),ZDELTA(51),ZDN(51,2)
     1               ,ZP,ZA
      COMMON/PEDATA/ NBODYS,XMIDB(51), NTHETB,THETB(11),CPBB(51,11)
     1         ,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH, CPB
     2         , SID(20)
      COMMON/WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
      COMMON/EXTRA/ ARAN, DSB1, DSB2, DS1, DS2, NB1,NB2,NS1,NS2,
     1          NTN, NXN, XNOSE
      COMMON/TRNSFR/ IOVR10, IOVR60, IOVR70
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/NSLOPE/ DRNDX(51,2), DZNDX(51,2)
C
      DIMENSION XNS1(2), XNS2(2), NS1(2), NS2(2), DS1(2), DS2(2)
      DIMENSION CPB(110), CPU(110), CPL(110)
      INTEGER TITLE
C
C
C        RESTART RUN IRW = 2
C
      IF ( IRW  .EQ. 2 ) GO TO 100
C
      IGOTO = IOVR70 + 1
      GO TO ( 100, 200, 300, 400, 500 ) , IGOTO
C
  100 CONTINUE
      CALL FORCES
      GO TO 600
C
  200 CONTINUE
      CALL OVERLAY(4HWANG, 7, 1)
      GO TO 600
C
  300 CONTINUE
      CALL OVERLAY(4HWANG, 7, 2)
      GO TO 600
C
  400 CONTINUE
      CALL OVERLAY(4HWANG, 7, 3)
      GO TO 600
C
  500 CONTINUE
      CALL OVERLAY(4HWANG, 7, 4)
C
  600 CONTINUE
C
C
      END
CDECK FORCES
      SUBROUTINE FORCES
C                                                                       00000020
C ..................................................................... 00000030
C     CONTROL ROUTINE FOR FORCES LINK                                   00000040
C ..................................................................... 00000050
```

```
C                                                                       00000060
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON /AVAR/  A(210),ACB(21),AEX(100),AWX(110),AREA(210),ARN(2),A00000090
     1               RWT(20),ALPHA(210),ALPHAB(210),ALPHAS(210),ALPHAT(100000100
     2               10),ALPHAW(110),ALPHAX(110),AWS(110),ABT(100),AN1N(00000110
     3               210,2),ALPHAA,ALPHAD,ARA,ARADEG,ARB,ARW,ARAS,AT,AAA00000120
      COMMON /BODS/ NUMBOD,XNOS(3),YNOS(3),ZNOS(3),NBSTAT(3)            00000130
      COMMON /BVAR/  B(210)                                             00000140
     1               ,BBCL,BBCD,BBCM                                    00000150
      COMMON/CVAR/C(210), CHORD(210),CL(210),CPNN(51,11,2)
     1               ,CLS(210)                                          00000170
     2               ,CDR,CASE,CPCALC,CBAR,CONSNT,CLBAR,CLX,CLM,CDM     00000180
      COMMON/DVAR/ D(210),DZDXB(51), DADEG,DARAD
      COMMON /IVAR/  IPOLAR                                             00000210
      COMMON /KVAR/  KASE,KONFIG,KPOLAR                                 00000220
      COMMON /NVAR/  NFMT(9),NROW(2)                                    00000230
     1         ,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,NTHETS,
     2         NXLE, NRG, NPOLR,  NCLX
      COMMON /PVAR/  POLAR                                              00000260
      COMMON/RVAR/ R(51), RN(51,2), RATIOX, RFAREA
      COMMON /SVAR/  SEMIS,SLC                                          00000290
      COMMON/TVAR/ TITLE(20), THETA(210), THETAB(11),THETAS(11), THETAN
     1         (11,2), TCL(11), TCD(11), TCM(11), THICK,TWIST
      COMMON /UVAR/  UBWT(100),UWBT(110),UWWT(110),UNCL(210,2)          00000330
      COMMON /VVAR/  VBWT(100),VWBT(110),VWWT(110),VNCL(210,2)          00000340
     1               ,VOUT                                              00000350
      COMMON /WVAR/  WBWT(100),WWBT(110),WWWT(110),WNCL(210,2)          00000360
      COMMON/XVAR/ XBAR(210), XC(210), XNI(2), XNXN(2), XNTN(2), XB(51),
     1         XN(51,2), XNNCD(2), XNNCL(2), XNNCM(2), XYZ(3), XCL(11),
     2         XCD(11), XNACEL, XP, XCPBAR
      COMMON /YVAR/  YBAR(210),YC(210),YNI(2)                           00000410
      COMMON/ZVAR/ ZBAR(210), ZC(210), ZNI(2), ZDELTA(51), ZDN(51,2)
     1               ,ZP,ZA                                             00000430
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1         ,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH
     2         ,CPB, SID(20)
      COMMON/WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
      COMMON/EXTRA/ ARAN, DSB1, DSB2, DS1, DS2, NB1, NB2, NS1, NS2,
     1         NTN, NXN, XNOSE
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/NSLOPE/ DRNDX(51,2), DZNDX(51,2)
C                                                                       00000440
      DIMENSION           XNS1(2),XNS2(2),NS1(2),NS2(2),DS1(2),DS2(2)  00000450
     1         , CPB(110), CPU(110), CPL(110)
      INTEGER TITLE                                                     00000460
C                                                                       00000470
C                                                                       00000480
C     DEFINE CONSTANTS AND READ INPUT DATA                              00000490
C     KONFIG = 1     WING ALONE CONFIGURATION                           00000500
C     KONFIG = 2     BODY ALONE CONFIGURATION                           00000510
C     KONFIG = 3     WING-BODY CONFIGURATION                            00000520
C                                                                       00000530
C     CASE = 1.      CALCULATE ALPHA, GIVEN CL                          00000540
C     CASE = 2.      CALCULATE CL, GIVEN ALPHA                          00000550
C     CASE = 3.      OPTIMIZE WING                                      00000560
C                                                                       00000570
C     CPCALC =0.     LINEAR CP                                          00000580
C     CPCALC =1.     NON-LINEAR CP                                      00000590
C     CPCALC =2.     EXACT CP                                           00000600
C                                                                       00000610
C     POLAR = 0.     POLARS NOT REQUESTED,OMIT DELTA ALPHA CARDS        00000620
C     POLAR = 1.     POLARS REQUESTED,INSERT DELTA ALPHA CARDS          00000630
C                                                                       00000640
```

```
C     THICK =0.      WING THICKNESS PRESSURES NOT TO BE ADDED         00000650
C     THICK =1.      WING THICKNESS PRESSURES TO BE ADDED             00000660
C                                                                     00000670
C     VOUT = 0.      VELOCITY COMPONENTS NOT TO BE PRINTED            00000680
C     VOUT = 1.      VELOCITY COMPONENTS TO BE PRINTED                00000690
C                                                                     00000700
C     XNACEL =0.     EFFECTS OF NACELS NOT INCLUDED                   00000710
C     XNACEL =X.     EFFECTS OF X PAIRS OF NACELLES, GEOMETRY EXPECTED 00000720
C     XNACEL =-X.    EFFECTS OF X PAIRS OF NACELLES, GEOMETRY OF SECOND 00000730
C                    PAIR SAME AS FIRST PAIR                          00000740
C                                                                     00000750
C     RFAREA =0.     WING REFERENCE AREA TO BE CALCULATED             00000760
C     RFAREA =X.     WING REFERENCE AREA TO BE SPECIFIED              00000770
C                                                                     00000780
C     XP,ZP          X AND Z COORDINATES OF POINT ABOUT WHICH MOMENTS 00000790
C                    ARE TO BE COMPUTED                               00000800
C                                                                     00000810
C     CBAR           REFERENCE CHORD LENGTH USED IN PITCHING          00000820
C                    MOMENT CALCULATIONS                              00000830
C                    CBAR =1. IF NOT SPECIFIED                        00000840
C                                                                     00000850
C     SEMIS          WING SEMI-SPAN USEDIN SPANWISE CD AND CL         00000860
C                    CALCULATIONS                                     00000870
C                    SEMIS =1. IF NOT SPECIFIED                       00000880
C                                                                     00000890
C                                                                     00000900
      KONFIG=KACE                                                     00000910
      NPANEL=NWING+NBODY                                              00000920
      NS=NBODY+1                                                      00000930
C                                                                     00000940
      CDR=57.2957795                                                  00000950
      IF ( IRW  .EQ. 2 ) GO TO 1111
      IF (KONFIG-2)1000,1010,1000                                     00000960
C                                                                     00000970
C     READ CONFIGURATION GEOMETRY                                     00000980
 1000 CONTINUE                                                        00000990
      READ  (NTAPEC) NBODY,NWING,XMACH,SYM,KACE                       00001000
      READ (NTAPEC) (J,XBAR(I),YBAR(I),ZBAR(I),XC(I),YC(I),ZC(I),AREA(I
     1),THETA(I),ALPHAS(I),CHORD(I),I=1,NPANEL),NRG,(NROW(I),I=1,NRG),RA00001020
     2TIOX                                                            00001030
      REWIND NTAPEC                                                   00001040
C                                                                     00001050
 1010 CONTINUE                                                        00001060
C                                                                     00001070
      NACEL=0                                                         00001100
C                                                                     00001110
C     READ CARD 5A                                                    00001120
      READ (NTAPEI,1700)CASE, CPCALC, POLAR, THICK, XNACEL, PINF, PUNCH,
     1          SOLID,WING,(SID(K),K=1,20),(IFORM(1),K=1,10)
      IPUNCH = IFIX( PUNCH )
      ISOLID = IFIX( SOLID )
      NWG = IFIX( WING )
      ANCEL=ABS(XNACEL)                                               00001140
      NACEL=ANCEL                                                     00001150
      NUMBOD=0                                                        00001160
      IF(NBODY.NE.0) NUMBOD=1                                         00001170
      XNOS(1)=0.                                                      00001171
      YNOS(1)=0.                                                      00001180
      ZNOS(1)=0.                                                      00001190
      NBSTAT(1)=NBODYS                                                00001200
      IPOLAR=0                                                        00001210
      POLAR = POLAR + 1
      IF (CASE)1020,1680,1020                                         00001220
 1020 CONTINUE                                                        00001230
C                                                                     00001240
```

```
C     RE'D CARD 6A                                                      00001250
      IF (NACEL)1120,1100,1030                                          00001270
 1030 DO 1090 N=1,NACEL                                                 00001280
      NDUM=NUMBOD+N                                                     00001290
C                                                                       00001300
C     READ CARD 6S1                                                     00001310
      READ(NTAPEI,1700) XNI(N),YNI(N),ZNI(N),XNXN(N),XNTN(N),ARN(N)     00001320
      XNOS(NDUM)=XNI(N)                                                 00001321
      YNOS(NDUM)=YNI(N)                                                 00001330
      ZNOS(NDUM)=ZNI(N)                                                 00001340
      IF(XNACEL)1040,1090,1050                                          00001350
 1040 IF(N-1)1060,1050,1060                                             00001360
 1050 NXN=XNXN(N)                                                       00001370
      NBSTAT(NDUM)=NXN                                                  00001380
      NTN=XNTN(N)                                                       00001390
C                                                                       00001400
C     READ CARD 6S2                                                     00001410
      READ(NTAPEI,1700) (XN(M,N),M=1,NXN)                               00001420
C     READ CARD 6S3                                                     00001430
      READ(NTAPEI,1700) (RN(M,N),M=1,NXN)                               00001440
C     READ CARD 6S4                                                     00001450
      READ(NTAPEI,1700) (ZDN(M,N),M=1,NXN)                              00001460
C     READ CARD 6S5                                                     00001470
      READ(NTAPEI,1700) (THETAN(M,N),M=1,NTN)                           00001480
C     READ CARD 6S6                                                     00001490
      READ (NTAPEI,1700) XNS1(N), XNS2(N), DS1(N), DS2(N), ( DRNDX(J,N),
     1               DZNDX(J,N), J=1,NXN )
      NS1(N)=XNS1(N)                                                    00001510
      NS2(N)=XNS2(N)                                                    00001520
      GO TO 1090                                                        00001530
C                                                                       00001540
 1060 DO 1070 M=1,NXN                                                   00001550
      XN(M,N)=XN(M,1)                                                   00001560
      RN(M,N)=RN(M,1)                                                   00001570
 1070 ZDN(M,N)=ZDN(M,1)                                                 00001580
C                                                                       00001590
      DO 1080 M=1,NTN                                                   00001600
 1080 THETAN(M,N)=THETAN(M,1)                                           00001610
C                                                                       00001620
      NS1(N)=NS1(1)                                                     00001630
      NS2(N)=NS2(1)                                                     00001640
      DS1(N)=DS1(1)                                                     00001650
      DS2(N)=DS2(1)                                                     00001660
C                                                                       00001670
 1090 CONTINUE                                                          00001680
      NUMBOD=NUMBOD+NACEL                                               00001690
 1100 DO 1110 N=1,2                                                     00001700
      XNNCL(N)=0.                                                       00001710
      XNNCD(N)=0.                                                       00001720
      XNNCM(N)=0.                                                       00001730
      DO 1110 J=1,NPANEL                                                00001740
      UNCL(J,N)=0.                                                      00001750
      VNCL(J,N)=0.                                                      00001760
      WNCL(J,N)=0.                                                      00001770
 1110 AN1N(J,N)=0.                                                      00001780
C                                                                       00001790
 1120 CONTINUE                                                          00001800
      IF (CBAR .EQ. 0.) CBAR=1.                                         00001810
      IF (SEMIS .EQ. 0.)SEMIS=1.                                        00001820
C                                                                       00001830
      IF (VOUT)1130,1150,1130                                           00001840
 1130 CONTINUE                                                          00001850
      DO 1140 J=1,9                                                     00001860
 1140 NFMT(J)=J                                                         00001870
      GO TO 1170                                                        00001880
```

```
 1150 CONTINUE
      DO 1160 J=1,9
 1160 NFMT(J)=0
 1170 CONTINUE
      KASE=CASE
C
 1180 IF (KONFIG-2)1200,1190,1200
 1190 RFAREA=1.0
      GO TO 1220
 1200 RFAREA=0.0
      DO 1210 J=NS,NPANEL
 1210 RFAREA=RFAREA+AREA(J)
 1220 CONTINUE
C
C     WRITE CASE DATA
      CALL INOUT(NTAPEO,KASE,CPCALC,POLAR,THICK,VOUT,XMACH,SEMIS,
     1XP,ZP,CBAR,RFAREA,SYM)
C
C
C     ANY CONFIGURATION
      IF (KONFIG .EQ. 1) GO TO 1280
C
C     BODY ALONE OR WING-BODY CONFIGURATION
      NROWB=NROW(1)
C
      IF (KONFIG-2)1240,1240,1250
C
C     BODY ALONE CONFIGURATION
 1240 READ (NTAPEC) NBODY,NWING,XMACH,SYM,KACE
      GO TO 1260
C
C     WING-BODY CONFIGURATION
 1250 CALL FSF(3,NTAPEC,IRR)
 1260 CONTINUE
      READ (NTAPEC) ALPHAA,DSB1,DSB2,XBS1,XBS2
      NB1 = XBS1
      NB2 = XBS2
      READ (NTAPEC) NBODYS,NXLE,ZA,(XB(I),I=1,NBODYS)
      IF (KONFIG .EQ. 2) GO TO 1270
      READ (NTAPEC) NTHETA,(THETAB(I),I=1,NTHETA)
C
C     INPUT BODY VARIABLES
 1270 READ (NTAPEC) NTHETB,(THETB(I),I=1,NTHETB)
C
C     READ CARD 7A
      READ ( NTAPEI,1700) ARB, DADEG
C     READ CARD 8A
      READ ( NTAPEC ) ( R(I), I=1,NBODYS )
C     READ CARD 9A
      READ ( NTAPEC ) ( ZDELTA(I), I=1,NBODYS )
      REWIND NTAPEC
C
      RBMAX = R(1)
      DO 1271 I=2,NBODYS
      RBMAX = AMAX1( RBMAX, R(I) )
 1271 CONTINUE
C
      IF ( KONFIG  .EQ. 2 ) RFAREA = 3.14159 * RBMAX**2
      IF ( KONFIG  .EQ. 2 ) WRITE (NTAPEO,1780)  RFAREA
C
      ARA=-ALPHAA+ARB/CDR
      WRITE (NTAPEO,1770) ZA
C
      IF (KONFIG .EQ. 2) GO TO 1290
C
```

```
C       WING-BODY CONFIGURATION
        CALL BCAM(NBODYS,NROWB,XB,ZDELTA,XC,DZDXB,ACB)
        CALL BTHICK(NBODYS,NBODY,NROWB,XB,R,XC,ALPHA,ALPHAS)
C
C       WING ALONE OR WING-BODY CONFIGURATION
 1280   CONTINUE
        NROWW=NROW(NRG)
        NCOLW=NWING/NROWW
C
        IF (KONFIG .EQ. 3) GO TO 1290
C
C       WING ALONE CONFIGURATION
        ALPHAA=0.
        ARA=0.
C
C       ANY CONFIGURATION
 1290   CONTINUE
        CALL FSF(7,NTAPEC,IRR)
        WRITE (NTAPEC) THICK,ARA
        CALL FOR EOF (NTAPEC)
        REWIND NTAPEC
        ALPHAD=ALPHAA*CDR
        ARADEG=ARA*CDR
        WRITE (NTAPEO,1750) ALPHAD
        WRITE (NTAPEO,1760) ARADEG
C
        IF (KONFIG.EQ.2) GO TO 1400
C
C       WING ALONE OR WING-BODY CONFIGURATION
C       INPUT WING VARIABLES
        GO TO (1300,1320,1370),KASE
C
C       READ CARD 10A
 1300   CONTINUE
        ICAM =2
        CALL READ(NTAPEI,NTAPEO,NTAPEC,NWING,CL(NS),KACE,ICAM,NROW)
        DO 1310 J=NS,NPANEL
 1310   CL(J)=-CL(J)
        GO TO 1390
C
C       READ CARD 11A
 1320   CONTINUE
        IF ( KONFIG  .EQ. 1 ) READ (NTAPEI, 1700) DUM, DADEG
        READ (NTAPEI,1700) ARW, TWIST
        IF (TWIST .NE. 0.) GO TO 1340
        DO 1330 J=1,NCOLW
 1330   ARWT(J)=0.0
        GO TO 1350
C
C       READ CARD 11AA
 1340   READ (NTAPEI,1700) (ARWT(I),I=1,NCOLW)
 1350   CONTINUE
        CALL FSF (2,NTAPEC,IRR)
C
C       READ CARD 12A
        ICAMBB = 2
        CALL READ(NTAPEI,NTAPEO,NTAPEC,NWING,ALPHA,KACE,ICAMBB,NROW)
        REWIND NTAPEC
        DO 1360 J=1,NWING
        JJ=(J-1)/NROWW+1
        JW=J+NBODY
        COST=COS(THETA(JW))
 1360   ALPHAW(J)=ALPHA(J)-ARA*COST-(ARW+ARWT(JJ))/CDR
        GO TO 1390
C
```

```
C       READ CARD 13A
 1370  READ (NTAPEI,1700) CONSNT,CLBAR,XCPBAR
       IF (CONSNT .NE. 0.) GO TO 1380
       WRITE (NTAPEO,1730) CLBAR
       GO TO 1390
 1380  WRITE (NTAPEO,1730) CLBAR
       WRITE (NTAPEO,1740) XCPBAR
 1390  CONTINUE
C
C      WING ALONE OR WING-BODY CONFIGURATION
       IF (THICK)1430,1400,1430
C
 1400  CONTINUE
       CALL FSF(8,NTAPEC,IRR)
       CALL FOR EOF (NTAPEC)
       REWIND NTAPEC
       IF (KONFIG.EQ.2) GO TO 1480
C
       DO 1420 J=1,NPANEL
 1420  ALPHA(J)=0.
       DO 1425 J=1,NWING
 1425  AWS(J)=0.
       IF (KONFIG.EQ.1) GO TO 1460
       GO TO 1440
C
C      WING ALONE OR WING-BODY CONFIGURATION
 1430  CONTINUE
C
C      READ CARD 14A
       CALL TVEL(A,B,C,D,ALPHAT,UBWT,VBWT,WBWT,UWWT,VWWT,WWWT,
      1ALPHA,CHORD,THICK,NROW)
       REWIND NTAPEA
       REWIND NTAPEB
C
       DO 1455 J=1,NWING
       JJ=J+NBODY
 1455  AWS(J)=ALPHA(JJ)
       IF (KONFIG .EQ. 1) GO TO 1460
C
C      WING-BODY CONFIGURATION
 1440  DO 1450 J=1,NBODY
       JJ=J-((J-1)/NROWB)*NROWB
       ALPHAB(J)=ALPHAS(J)-(ARA    -ACB(JJ))*COS(THETA(J))
       ABX(J)=-ALPHA(J)
       ABT(J)=ABX(J)
 1450  CONTINUE
C
C      ANY CONFIGURATION
 1460  CONTINUE
       DO 1470 J=1,NWING
       UWBT(J)=0.
       VWBT(J)=0.
       WWBT(J)=0.
 1470  ALPHAX(J)=0.
C
 1480  CONTINUE
       CALL FSF(9,NTAPEC,IRR)
       WRITE(NTAPEC) NUMBOD,(YNOS(I),ZNOS(I),NBSTAT(I),I=1,NUMBOD)
       IF ( KONFIG  .EQ. 1 ) GO TO 1620
C
C      BODY ALONE OR WING-BODY CONFIGURATION
C
 1490  XNOSE = XB(1)
       IF(XMACH-1.)1500,1620,1510
 1500  CALL OVERLAY(4HWANG, 7, 5)
```

```
      GO TO 1560
C
 1510 CALL OVERLAY(4HWANG, 7 , 6)
C
 1560 CONTINUE                                                          00003870
      IF(NBODY)1590,1590,1570                                           00003880
 1570 DO 1580 J=1,NBODY                                                 00003890
 1580 ABX(J)=ABT(J)-AN1N(J,1)-AN1N(J,2)                                 00003900
 1590 IF(NWING)1620,1620,1600                                           00003910
 1600 DO 1610 J=1,NWING                                                 00003920
      JJ=J+NBODY                                                        00003930
 1610 ALPHAX(J)=ALPHAX(J)+AN1N(JJ,1)+AN1N(JJ,2)                         00003940
 1620 CONTINUE                                                          00003950
      IF (NWING.LE.0) GO TO 1628                                        00003951
      DO 1625 J=1,NWING                                                 00003952
1625  ALPHAX(J)=ALPHAX(J)+AWS(J)                                        00003953
1628  CONTINUE                                                          00003954
      CALL FOR EOF(NTAPEC)                                              00003960
      REWIND NTAPEC                                                     00003970
C                                                                       00003980
C     ANY CONFIGURATION                                                 00003990
C
      CALL OVERLAY(4HWANG, 7, 7)
C
 1111 CONTINUE
      IF ( IRW  .GT. 0 ) CALL OVERLAY(4HWANG, 7, 11)
C
      IF ( KONFIG  .LT. 3 ) GO TO 1800
C
C        ADDING PRESSURE DUE TO THE PRESENCE OF WING TO BODY PANEL
C
      CALL OVERLAY(4HWANG, 7, 9)
C
 1800 CONTINUE
C
      NSYM = 1
C
      GO TO ( 1820, 1810, 1810 ) , KONFIG
C
C
C              INTERPOLATION OF PRESSURE
C
 1810 CONTINUE
C
C              BODY
C
      CALL OVERLAY(4HWANG,7, 8)
C
      IF ( KONFIG  .EQ.  3 ) NSYM = NSYM + 1
C
      IF ( KONFIG  .EQ. 2 ) GO TO 1840
C
 1820 CONTINUE
C
C              WING
C
      CALL OVERLAY (4HWANG, 7, 10)
C
 1840 CONTINUE
C
      NPOLAR = 1
C
C     ANY CONFIGURATION                                                 00004010
C     READ ADDITIONAL POLARS                                            00004020
      IF ( POLAR ) 1630, 1680, 1630
```

```
C
 1630 CONTINUE
C
C
 1640 CONTINUE
      IPOLAR=1
      DARAD=DADEG/CDR
      ARA=ARA+DARAD
      ARADEG=ARA*CDR
      WRITE (NTAPEO,1710) TITLE
      WRITE (NTAPEO,1720) DADEG
      WRITE (NTAPEO,1760) ARADEG
C
      IF (KONFIG .EQ. 2) GO TO 1670
C
C     WING ALONE OR WING-BODY CONFIGURATION
      DO 1650 I=1,NWING
      JW=I+NBODY
      COST=COS(THETA(JW))
1650  ALPHAW(I)=ALPHAW(I)-DARAD*COST
C
      IF (KONFIG .EQ. 1) GO TO 1670
C
C     WING-BODY CONFIGURATION
      DO 1660 I=1,NBODY
 1660 ALPHAB(I)=ALPHAB(I)-DARAD*COS(THETA(I))
C
C     ANY CONFIGURATION
 1670 CONTINUE
C
      IF(KONFIG-2)1460,1480,1480
C
 1680 CONTINUE
C
C----------------
C -------------------
C
C
C
 1690 FORMAT(20A4)
 1700 FORMAT(7F10.0)
 1710 FORMAT(1H1,20A4,30X,2A4)
 1720 FORMAT(1HO,23HDELTA ANGLE OF ATTACK =,F10.4,2X,4HDEG.)
 1730 FORMAT(1HO,27HWING OPTIMIZED FOR CL BAR =,F10.4)
 1740 FORMAT(1H , 18X,9HXCP BAR =,F10.4)
 1750 FORMAT(1HO,56HINCLINATION OF BODY AXIS WITH RESPECT TO DEFINING AX
     1IS =,F10.4,2X,4HDEG.)
 1760 FORMAT(1HO,43HANGLE OF ATTACK WITH RESPECT TO BODY AXIS =,F10.4,
     12X,4HDEG.)
 1770 FORMAT(1HO,38HHEIGHT OF WING PLANE ABOVE BODY AXIS =,F10.4)
 1780 FORMAT (1HO,*BODY REFERENCE AREA   =   *,F10.4)
      RETURN
      END
CDECK TVEL
      SUBROUTINE TVEL(A,B,C,D,ALPHAT,UBWT,VBWT,WBWT,UWWT,VWWT,WWWT,AN,
     1CHORD,THICK,NROW)
C
C ......................................................................
C     COMPUTE VELOCITY COMPONENTS INDUCED BY WING SOURCES
C ......................................................................
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /PANVEL/ VSS(220),WS(110)
C
```

```
      INTEGER TSL
      REAL ATLE(20)
      DIMENSION A(1),B(1),C(1),D(1),ALPHAT(1),UBWT(1),VBWT(1),WBWT(1),
     1UWWT(1),VWWT(1),WWWT(1),AN(1),NROW(1),CHORD(1)
C
C
      DATA TSL /4HGIVE/
C
      NS=NBODY+1
      NPANEL=NBODY+NWING
      IF (NBODY .EQ. 0) GO TO 1010
      DO 1000 J=1,NBODY
      UBWT(J)=0.
      VBWT(J)=0.
      WBWT(J)=0.
      AN(J)=0.
 1000 CONTINUE
 1010 DO 1020 J=1,NWING
      UWWT(J)=0.
      VWWT(J)=0.
      WWWT(J)=0.
      JJ=J+NBODY
      AN(JJ)=0.0
 1020 CONTINUE
C
      GO TO (1030,1180,1040),KACE
C
 1030 NC=NWING/NROW(1)
      NR=NROW(1)
      GO TO 1050
 1040 NC=NWING/NROW(2)
      NR=NROW(2)
 1050 N=0
      NWINC=NWING+NC
      IF ( THICK  .LT. 1.0 ) GO TO 1080
C
C     INPUT THICKNESS SLOPES
      K = -NR + 1
      DO 1070 I=1,NC
       K = K+NR
       KK = K+NR - 1
      READ (NTAPEI,1190) ATLE(I),(ALPHAT(J),J=K,KK)
 1070 CONTINUE
      GO TO 1110
C
C     USE PREVIOUSLY CALCULATED PANEL SLOPES
C     AND COMPUTE LEADING EDGE SLOPES
 1080 CALL FSF(1,NTAPEC,IRR)
      READ (NTAPEC) (ALPHAT(I),I=1,NWING)
      REWIND NTAPEC
      NR1=NR-1
C
      DO 1100 J=1,NC
      J1=(J-1)*NR
      JN=NBODY+J1
      AT=-ALPHAT(J1+1)
      DO 1090 I=1,NR1
      IJ=J1+I
      IJN=NBODY+IJ
      AT=AT-(ALPHAT(IJ)+ALPHAT(IJ+1))*CHORD(IJN+1)/
     1CHORD(JN+1)
 1090 CONTINUE
      ATLE(J)=AT
 1100 CONTINUE
C
```


MCDONNELL DOUGLAS

```
1110 CONTINUE                                                           00000750
     CALL FSF(8,NTAPEC,IRR)                                             00000760
     WRITE (NTAPEC) NWINC                                               00000770
     DO 1118 J=1,NC                                                     00000780
     J1=(J-1)*NR                                                        00000781
     DO 1116 I=1,NR                                                     00000782
     IJ=I+J1                                                            00000783
     IF(I.EQ.1) GO TO 1114                                              00000784
     WS(IJ)=(ALPHAT(IJ-1)+ALPHAT(IJ))*.5                                00000785
     GO TO 1116                                                         00000786
1114 WS(IJ)=(ATLE(J)+ALPHAT(IJ))*.5                                     00000787
1116 CONTINUE                                                           00000788
1118 CONTINUE                                                           00000789
     DO 1170J=1,NWINC                                                   00000790
     READ (NTAPEA) (D(I),I=1,NPANEL)                                    00000800
     READ (NTAPEB) (A(I),B(I),C(I),I=1,NPANEL)                          00000810
     JT=J-N*(NR+1)                                                      00000820
     IF (JT)1130,1130,1120                                              00000830
1120 N=N+1                                                              00000840
     WT=ATLE(N)                                                         00000850
     GO TO 1140                                                         00000860
1130 JN=J-N                                                             00000870
     WT=ALPHAT(JN)                                                      00000880
1140 CONTINUE                                                           00000890
     WRITE (NTAPEC) WT                                                  00000900
     DO 1160 I=1,NPANEL                                                 00000910
     IF (I .GT. NBODY) GO TO 1150                                       00000920
     UBWT(I)=UBWT(I)+A(I)*WT                                            00000930
     VBWT(I)=VBWT(I)+B(I)*WT                                            00000940
     WBWT(I)=WBWT(I)+C(I)*WT                                            00000950
     AN(I)=AN(I)+D(I)*WT                                                00000960
     GO TO 1160                                                         00000970
1150 IN=I-NBODY                                                         00000980
     UWWT(IN)=UWWT(IN)+A(I)*WT                                          00000990
     VWWT(IN)=VWWT(IN)+B(I)*WT                                          00001000
     WWWT(IN)=WWWT(IN)+C(I)*WT                                          00001010
     AN(I)=AN(I)+D(I)*WT                                                00001011
1160 CONTINUE                                                           00001020
1170 CONTINUE                                                           00001030
C                                                                       00001040
1180 CONTINUE                                                           00001050
     CALL FOR EOF (NTAPEC)                                              00001060
     REWIND NTAPEC                                                      00001070
C                                                                       00001080
C ------------------                                                    00001090
     RETURN                                                             00001100
C ------------------                                                    00001110
C                                                                       00001120
1190 FORMAT (7F10.0)                                                    00001130
     END                                                                       -
CDECK READ
     SUBROUTINE READ(NTAPEI,NTAPEO,NTAPEC,NM,S,KACE,ICAMBR,NROW)
C                                                                       READ1010
C ..................................................................... READ1020
C    CONTROLS DATA READ IN OPTION FOR SPECIFYING WING CAMBER AND        READ1030
C    PRESSURE DISTRIBUTION, ALSO BODY CAMBER                            READ1040
C ..................................................................... READ1050
C                                                                       READ1060
     DIMENSION NROW(2)                                                  READ1070
C                                                                       READ1080
     REAL S(1)                                                          READ1090
     INTEGER DICT(2)                                                    READ1100
C                                                                       READ1110
     DATA DICT(1)/4HCONS/                                               READ1120
     DATA DICT(2)/4HGIVE/                                               READ1130
```

```
C                                                                       READ1140
C        ICOMBR = 1  CONSTANT CAMBER, 2  COMPUTED, 3  READ IN
      GO TO ( 1060, 1080, 1000), ICAMBR
 1000 CONTINUE                                                          READ1180
      IF (KACE .NE. 0) GO TO 1010                                       READ1190
      READ (NTAPEI,1100) (S(I),I=1,NM)                                  READ1200
      GO TO 1090                                                        READ1210
C                                                                       READ1220
 1010 GO TO (1020,1090,1030), KACE                                      READ1230
 1020 NC = NM / NROW(1)                                                 READ1240
      NR = NROW(1)                                                      READ1250
      GO TO 1040                                                        READ1260
 1030 NC = NM / NROW(2)                                                 READ1270
      NR = NROW(2)                                                      READ1280
 1040 K = -NR + 1                                                       READ1290
      DO 1050 I=1,NC                                                    READ1300
       K = K+NR                                                         READ1310
       KK = K+NR - 1                                                    READ1320
       READ (NTAPEI,1100) (S(J),J=K,KK)                                 READ1330
 1050 CONTINUE                                                          READ1340
       GO TO 1090                                                       READ1350
C                                                                       READ1360
 1060 CONTINUE                                                          READ1370
      READ (NTAPEI,1100) SCONST                                         READ1380
      DO 1070 I=1,NM                                                    READ1390
 1070 S(I)=SCONST                                                       READ1400
      GO TO 1090                                                        READ1410
 1080 CONTINUE                                                          READ1420
      READ (NTAPEC) (S(I),I=1,NM)                                       READ1430
 1090 CONTINUE                                                          READ1440
C                                                                       READ1450
C ------------------                                                    READ1460
      RETURN                                                            READ1470
C ------------------                                                    READ1480
 1100 FORMAT(7F10.0)                                                    READ1490
      END                                                                  -
CDECK BTHICK
      SUBROUTINE BTHICK(NBODYS,NBODY,NROWB,XB,R,XC,ALPHA,THETA)         BTHI1000
C                                                                       BTHI1010
C ..................................................................... BTHI1020
C     COMPUTES BODY THICKNESS SLOPES FOR GIVEN BODY RADII DISTRIBUTION  BTHI1030
C ..................................................................... BTHI1040
C                                                                       BTHI1050
      DIMENSION XB(1),R(1),XC(1),ALPHA(1),THETA(1)                      BTHI1060
C                                                                       BTHI1070
      NB=NBODYS-1                                                       BTHI1080
      NCOLB=NBODY/NROWB                                                 BTHI1090
C                                                                       BTHI1100
      DO 1000 I=1,NB                                                    BTHI1110
 1000 ALPHA(I+1)=(R(I+1)-R(I))/(XB(I+1)-XB(I))                          BTHI1120
      ALPHA(1)=ALPHA(2)                                                 BTHI1130
      DO 1030 J=1,NROWB                                                 BTHI1140
      DO 1010 I=1,NB                                                    BTHI1150
      IF (XB(I)-XC(J))1010,1020,1020                                    BTHI1160
 1010 CONTINUE                                                          BTHI1170
      I=NBODYS                                                          BTHI1180
 1020 THETA(J)=ALPHA(I-1)+(XC(J)-XB(I-1))/(XB(I)-XB(I-1))*(ALPHA(I)-    BTHI1190
     1ALPHA(I-1))                                                       BTHI1200
 1030 CONTINUE                                                          BTHI1210
C                                                                       BTHI1220
      DO 1040 J=1,NROWB                                                 BTHI1230
      DO 1040 I=2,NCOLB                                                 BTHI1240
      IJ=(I-1)*NROWB+J                                                  BTHI1250
      THETA(IJ)=THETA(J)                                                BTHI1260
 1040 CONTINUE                                                          BTHI1270
```

```
C                                                                       BTHI1280
C ------------------                                                    BTHI1290
      RETURN                                                            BTHI1300
C ------------------                                                    BTHI1310
      END                                                               -
CDECK BCAM
      SUBROUTINE BCAM(NBODYS,NROWB,XB,ZDELTA,XC,DZDXB,ACB)              BCAM1000
C                                                                       BCAM1010
C ..................................................................... BCAM1020
C     COMPUTES BODY CAMBER SLOPES, GIVEN BODY CAMBER SHAPE              BCAM1030
C ..................................................................... BCAM1040
C                                                                       BCAM1050
      DIMENSION XB(1),ZDELTA(1),XC(1),DZDXB(1),ACB(1)                   BCAM1060
C                                                                       BCAM1070
      NB=NBODYS-1                                                       BCAM1080
      DO 1000 I=1,NB                                                    BCAM1090
 1000 DZDXB(I+1)=(ZDELTA(I+1)-ZDELTA(I))/(XB(I+1)-XB(I))                BCAM1100
      DZDXB(1)=DZDXB(2)                                                 BCAM1110
C                                                                       BCAM1120
      DO 1030 J=1,NROWB                                                 BCAM1130
      DO 1010 I=1,NB                                                    BCAM1140
      IF (XB(I)-XC(J))1010,1020,1020                                    BCAM1150
 1010 CONTINUE                                                          BCAM1160
 1020 ACB(J)=DZDXB(I-1)+(XC(J)-XB(I-1))/(XB(I)-XB(I-1))*(DZDXB(I)-DZDXB(BCAM1170
     1I-1))                                                             BCAM1180
 1030 CONTINUE                                                          BCAM1190
C                                                                       BCAM1200
C ------------------                                                    BCAM1210
      RETURN                                                            BCAM1220
C ------------------                                                    BCAM1230
      END                                                               -
CDECK INOUT
      SUBROUTINE INOUT(NTAPEO,KASE,CPCALC,POLAR,THICK,VOUT,XMACH,SEMIS, INOU1000
     1XP,ZP,CBAR,RFAREA,SYM)                                            INOU1010
C                                                                       INOU1020
C ..................................................................... INOU1030
C     PRINTS OUT ALL INPUT AERODYNAMIC DATA                             INOU1040
C ..................................................................... INOU1050
C                                                                       INOU1060
C                                                                       INOU1070
      WRITE (NTAPEO,1200)                                               INOU1080
      IF (SYM)1010,1000,1010                                            INOU1090
 1000 WRITE (NTAPEO,1210)                                               INOU1100
      GO TO 1020                                                        INOU1110
 1010 WRITE (NTAPEO,1220)                                               INOU1120
 1020 CONTINUE                                                          INOU1130
C                                                                       INOU1140
      GO TO (1030,1040,1050),KASE                                       INOU1150
 1030 WRITE (NTAPEO,1230)                                               INOU1160
      GO TO 1060                                                        INOU1170
 1040 WRITE (NTAPEO,1240)                                               INOU1180
      GO TO 1060                                                        INOU1190
 1050 WRITE (NTAPEO,1250)                                               INOU1200
 1060 CONTINUE                                                          INOU1210
C                                                                       INOU1220
      IF (CPCALC-1.)1070,1080,1090                                      INOU1230
 1070 WRITE (NTAPEO,1260)                                               INOU1240
      GO TO 1100                                                        INOU1250
 1080 WRITE (NTAPEO,1270)                                               INOU1260
      GO TO 1100                                                        INOU1270
 1090 WRITE (NTAPEO,1280)                                               INOU1280
 1100 CONTINUE                                                          INOU1290
C                                                                       INOU1300
      IF (POLAR)1120,1110,1120                                          INOU1310
 1110 WRITE (NTAPEO,1290)                                               INOU1320
```

```
      GO TO 1130
 1120 WRITE (NTAPEO,1300)
 1130 CONTINUE
C
      IF (THICK)1150,1140,1150
 1140 WRITE (NTAPEO,1310)
      GO TO 1160
 1150 WRITE (NTAPEO,1320)
 1160 CONTINUE
C
      IF (VOUT)1180,1170,1180
 1170 WRITE (NTAPEO,1330)
      GO TO 1190
 1180 WRITE (NTAPEO,1340)
 1190 CONTINUE
C
      WRITE (NTAPEO,1380) XMACH
      WRITE (NTAPEO,1370) XP,ZP,CBAR
      WRITE (NTAPEO,1360) RFAREA
      WRITE (NTAPEO,1350) SEMIS
C
C
C
C ------------------
      RETURN
C ------------------
 1200 FORMAT(1HO//30H DESCRIPTION OF CASE REQUESTED)
 1210 FORMAT(1HO/85H UNSYMMETRICAL CONFIGURATION - PANELS LOCATED ONLY O
     1N ONE SIDE OF X-Z PLANE(SYM = 0.))
 1220 FORMAT(1HO/80H SYMMETRICAL CONFIGURATION - PANELS LOCATED ON BOTH
     1SIDES OF X-Z PLANE(SYM = 1.))
 1230 FORMAT(1HO,9HCASE = 1.,10X,35HCALCULATE WING SHAPE, GIVEN WING CL)
 1240 FORMAT(1HO,9HCASE = 2.,10X,25HCALCULATE CL, GIVEN SHAPE)
 1250 FORMAT(1HO,9HCASE = 3.,10X,19HOPTIMIZE WING SHAPE)
 1260 FORMAT(1HO,11HCPCALC = 0.,8X,9HLINEAR CP)
 1270 FORMAT(1HO,11HCPCALC = 1.,8X,13HNON-LINEAR CP)
 1280 FORMAT(1HO,11HCPCALC = 2.,8X,8HEXACT CP)
 1290 FORMAT(1HO,10HPOLAR = 0. ,9X,20HPOLARS NOT REQUESTED)
 1300 FORMAT(1HO,10HPOLAR = 1. ,9X,16HPOLARS REQUESTED)
 1310 FORMAT(1HO,10HTHICK = 0.,9X,40HWING THICKNESS PRESSURES NOT TO BE
     1ADDED)
 1320 FORMAT(1HO,10HTHICK = 1.,9X,36HWING THICKNESS PRESSURES TO BE ADDE
     1D)
 1330 FORMAT(1HO,9HVOUT = 0.,10X,37HVELOCITY COMPONENTS NOT TO BE PRINTE
     1D)
 1340 FORMAT(1HO,9HVOUT = 1.,10X,33HVELOCITY COMPONENTS TO BE PRINTED)
 1350 FORMAT (1HO,17HWING SEMI-SPAN = ,F10.4)
 1360 FORMAT(1HO,21HWING REFERENCE AREA =,F10.4)
 1370 FORMAT(1HO,48HPOINT ABOUT WHICH THE MOMENTS ARE TO BE COMPUTED/16H
     1  X-COORDINATE =,F10.4/16H  Z-COORDINATE =,F10.4//
     225H REFERENCE CHORD LENGTH =,F10.4)
 1380 FORMAT(1HO,13HMACH NUMBER =,F10.4)
      END
CDECK CENTRO
      SUBROUTINE CENTRO( X, XC, NG, NGRIDP, NELM, NE)
C
C
C              COMPUTES CENTROID COORDINATES OF FINITE ELEMENT,
C           TRIANGULAR OR QUADRILATERAL.
C
      DIMENSION X( 3, NGRIDP), XC( 3, NE )
      DIMENSION NG(NGRIDP), NELM(NE)
      DATA IWORD1 / 5HCQUAD /
C
C              READ FINITE ELEMENT GRID SPECIFICATION FROM BING OUTPUTS.
```

```
C
      DO 200 I= 1, NE
      READ (5, 901) ICHECK, ID, IE, ND, NG1, NG2, NG3, FG4, DM
      NELM(I) = IE
      NG4 = FG4
C
C              SEARCH GRID POINT NUMBER
      ICOUNT = 1
      DO 110 K=1,NGRIDP
      IF ( NG(K)   .EQ. NG1 ) GO TO 101
      IF ( NG(K)   .EQ. NG2 ) GO TO 102
      IF ( NG(K)   .EQ. NG3 ) GO TO 103
      IF ( NG(K)   .EQ. NG4 ) GO TO 104
      GO TO 110
  101 I1 = K
      GO TO 109
  102 I2 = K
      GO TO 109
  103 I3 = K
      GO TO 109
  104 I4 = K
  109 IF ( ICOUNT  .EQ.  4 ) GO TO 150
      IF (  ICOUNT  .EQ. 3  .AND.  ICHECK  .NE.  IWORD1 ) GO TO 111
      ICOUNT = ICOUNT + 1
  110 CONTINUE
C
C              TRIANGULAR ELEMENT
C
  111 CONTINUE
      DO 120  J= 1, 3
      XC( J, I) = ( X( J, I1) + X( J, I2) + X( J, I3) ) / 3.0
  120 CONTINUE
      GO TO 200
C
  150 CONTINUE
C
C              QUADRILATERAL ELEMENT
C
      X1 = ( X( 1, I1 ) + X( 1, I2 ) + X( 1, I3 ) ) / 3.0
      Y1 = ( X( 2, I1 ) + X( 2, I2 ) + X( 2, I3 ) ) / 3.0
      Z1 = ( X( 3, I1 ) + X( 3, I2 ) + X( 3, I3 ) ) / 3.0
      X2 = ( X( 1, I3 ) + X( 1, I4 ) + X( 1, I1 ) ) / 3.0
      Y2 = ( X( 2, I3 ) + X( 2, I4 ) + X( 2, I1 ) ) / 3.0
      Z2 = ( X( 3, I3 ) + X( 3, I4 ) + X( 3, I1 ) ) / 3.0
C
      S12 = SQRT( ( X(1,I1) - X(1,I2) )**2 + ( X(2,I1) - X(2,I2) )**2
     1          + ( X(3,I1) - X(3,I2) )**2 )
      S23 = SQRT( ( X(1,I2) - X(1,I3) )**2 + ( X(2,I2) - X(2,I3) )**2
     1          + ( X(3,I2) - X(3,I3) )**2 )
      S31 = SQRT( ( X(1,I3) - X(1,I1) )**2 + ( X(2,I3) - X(2,I1) )**2
     1          + ( X(3,I3) - X(3,I1) )**2 )
      S34 = SQRT( ( X(1,I3) - X(1,I4) )**2 + ( X(2,I3) - X(2,I4) )**2
     1          + ( X(3,I3) - X(3,I4) )**2 )
      S41 = SQRT( ( X(1,I4) - X(1,I1) )**2 + ( X(2,I4) - X(2,I1) )**2
     1          + ( X(3,I4) - X(3,I1) )**2 )
      ST123 = 0.5 * ( S12 + S23 + S31 )
      ST341 = 0.5 * ( S31 + S34 + S41 )
      A123 = SQRT( ST123 * ( ST123-S12) * ( ST123-S23) * (ST123-S31) )
      A341 = SQRT( ST341*( ST341-S31)*(ST341-S34)*(ST341-S41) )
      AT = A123 + A341
      XC(1,I) = ( X1*A123 + X2*A341 ) / AT
      XC(2,I) = ( Y1*A123 + Y2*A341 ) / AT
      XC(3,I) = ( Z1*A123 + Z2*A341 ) / AT
C
  200 CONTINUE
```

```
C
  901 FORMAT ( A5, A3, 5I8, F8.0, F8.4 )
C
      RETURN
      END
CDECK CONVT
      SUBROUTINE CONVT( X, NG, NGRIDP, NSYM, ISYM)
C
C
C              CONVERT GRID POINT COORDINATES TO THE RECTANGULAR COORD.
C
      DIMENSION X( 3, NGRIDP), NG(NGRIDP), ISYM(20)
C
      DR = 0.0174532925
C
      DO 250  I = 1, NGRIDP
C
C              READ OUTPUT OF BING  FINITE ELEMENT GRID POINT COORDINATE
C
      READ (5, 901) CHECK,IDUM,NG(I), NS, X1, X2, X3, ND
      R = X1
      THETA = X2
      PHI = X3
      NGO = ISYM(NSYM) + 1
      GO TO ( 100, 150, 200 ) , NGO
C
C              RECTANGULAR COORD.
C
  100 CONTINUE
      X( 1, I) = X1
      X( 2, I) = X2
      X( 3, I) = X3
      GO TO 250
C              CYLINDRICAL COORD.
C
  150 CONTINUE
      X( 1, I) = R * COS( THETA * DR )
      X( 2, I) = R * SIN( THETA * DR )
      X( 3, I) = X3
      GO TO 250
C
C              SPHERICAL COORD.
C
  200 CONTINUE
      X( 1, I) = R * SIN( THETA*DR ) * COS( PHI * DR )
      X( 2, I) = R * SIN( THETA*DR ) * SIN( PHI * DR )
      X( 3, I) = R * COS( THETA*DR )
C
  250 CONTINUE
C
  901 FORMAT ( A5,A3, 2I8, 3F8.2, I8 )
C
      RETURN
      END
CDECK COORDT
      SUBROUTINE COORDT( X, NG)
C
C
C              TRANSFORM COORDINATES OF THE GRID POINT OF FINITE ELEMENT
C              TO THE AERO CODE SYSTEM USING THE TRANSFORM MATRIX TFM
C
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      DIMENSION X(3,NG), TX(3)
C
      NT = NTRANS(NSYM)
```

```
      DO 300 I=1, NG
C
      DO 210 ITRAN = 1, NT
C
      DO 200 J=1,3
C
      SUM = 0.0
      DO 150 K=1, 3
      JK = K + ( J-1) * 3
      SUM = SUM + X( K,I) * BTFM( JK, ITRAN, NSYM)
  150 CONTINUE
      TX(J) = SUM + VA( J, ITRAN, NSYM)
  200 CONTINUE
      DO 205  M=1,3
      X( M, I) = TX( M )
  205 CONTINUE
C
  210 CONTINUE
C
  300 CONTINUE
C
      RETURN
      END
CDECK,SURFIT
      SUBROUTINE  SURFIT( X, Y, N3, B )
C
C        COMPUTES COEFFICIENTS FOR SURFACE SPLINE.
C        N  TOTAL NUMBER OF INPUT POINTS, CAN BE RANDOMLY LOCATED.
C        JEQ  FOR EQUATION NUMBERING
C        MUK  FOR COEFFICIENT NUMBERING
C
C        CF  ARE THE COEFFICINT MATRIX OF EQUATION
C              W(X,Y) = B(1) + B(2)*X + B(3)*Y + SUM OF B(I) * R(I)**2 *
C              LN( R(I)**2)
C
      COMMON/REPEAT/ IREPT
      DIMENSION IPIV(115), INXR(115), INXC(115), CF(115,115), TM(115)
      DIMENSION X(N3), Y(N3), B(N3)
C
      N4 = 115
      N2 = N3 - 1
      N1 = N3 - 2
      N = N3 - 3
C
C        BYPASS MATRIX INVERSION FOR THE CASE WHERE THE DATA POINTS (X,
C        Y )  ARE IDENTICAL FOR INTERPOLATION
C
C
  101 CONTINUE
      DO 100 JEQ = 1, N
      CF( JEQ, 1) = 1.0
      CF ( JEQ, 2 ) =   X( JEQ )
      CF ( JEQ, 3 ) =   Y( JEQ )
      DO 90 MUK = 1, N
      MUK1 = MUK + 3
      IF ( MUK  .EQ.  JEQ ) GO TO 90
      RJM = ( X( MUK ) - X( JEQ ) )**2 + ( Y( MUK ) - Y( JEQ ) )**2
      CF( JEQ, MUK1 ) = RJM * ALOG( RJM )
   90 CONTINUE
  100 CONTINUE
C
      DO 150 K=1,3
      CF( N1, K) = 0.0
      CF( N2, K) = 0.0
      CF( N3, K) = 0.0
```

```
  150 CONTINUE
      DO 200  MUK =1,N
      MUK1 = MUK + 3
      CF( N1, MUK1) = 1.0
      CF ( N2, MUK1) =   X( MUK )
      CF( N3, MUK1 ) =   Y( MUK )
  200 CONTINUE
C
C
      IA = N3
      CALL INVERT( GOT, CF, IA, IPIV, INXR, INXC, N4)
C
  250 CONTINUE
      B(N1) = B(N2) = B(N3) = 0.0
C
      DO 300 I=1,N3
      SUM = 0.0
      DO 290 J=1,N3
      SUM = SUM + CF(I, J) * B(J)
  290 CONTINUE
      TM(I) = SUM
  300 CONTINUE
C
      DO 400 I=1,N3
      B(I) = TM(I)
  400 CONTINUE
C
      RETURN
      END
CDECK INVERT
      SUBROUTINE INVERT (GOTON, A, IA, IPIVOT, INDXR, INDXC, NROWS)
C                                                                      INVE1010
C ...................................................................... INVE1020
C                                                                      INVE1030
C ...................................................................... INVE1040
C                                                                      INVE1050
      REAL A(NROWS, NROWS), PIVOT, T                                   INVE1060
      INTEGER IPIVOT(IA), INDXR(IA), INDXC(IA)                         INVE1070
C                                                                      INVE1080
C                                                                      INVE1090
      GOTON = 1
      N = IA                                                           INVE1100
      DO 1000 J=1, N                                                   INVE1110
 1000 IPIVOT(J) = 0                                                    INVE1120
      DO 1090 I=1, N                                                   INVE1130
      T = 0.0                                                          INVE1140
      DO 1020 J=1, N                                                   INVE1150
      IF (IPIVOT(J).EQ.1) GO TO 1020                                   INVE1160
C                                                                      INVE1170
      DO 1010 K=1, N                                                   INVE1180
      IF (IPIVOT(K).EQ.1) GO TO 1010                                   INVE1190
      IF ( .NOT. (ABS(A(J,K)) -ABS(T) .GT. 0.0) ) GO TO 1010           INVE1200
      IROW = J                                                         INVE1210
      ICOL = K                                                         INVE1220
      T = A(J,K)                                                       INVE1230
 1010 CONTINUE                                                         INVE1240
 1020 CONTINUE                                                         INVE1250
      IPIVOT(ICOL) = IPIVOT(ICOL)+1                                    INVE1260
      IF (IROW.EQ.ICOL) GO TO 1040                                     INVE1270
      DO 1030 L=1, N                                                   INVE1280
      T = A(IROW,L)                                                    INVE1290
      A(IROW,L) = A(ICOL,L)                                            INVE1300
 1030 A(ICOL,L) = T                                                    INVE1310
 1040 INDXR(I) = IROW                                                  INVE1320
      INDXC(I) = ICOL                                                  INVE1330
```

```
      PIVOT = A(ICOL,ICOL)                                              INVE1340
      IF (PIVOT)1050,1120,1050                                          INVE1350
 1050 A(ICOL,ICOL) = 1.0                                                INVE1360
      DO 1060 L=1, N                                                    INVE1370
 1060 A(ICOL,L) = A(ICOL,L)/PIVOT                                       INVE1380
      DO 1080 L=1, N                                                    INVE1390
      IF (L.EQ.ICOL) GO TO 1080                                         INVE1400
      T = A(L,ICOL)                                                     INVE1410
      A(L,ICOL) = 0.0                                                   INVE1420
      DO 1070 M=1, N                                                    INVE1430
 1070 A(L,M) = A(L,M)-A(ICOL,M)*T                                       INVE1440
 1080 CONTINUE                                                          INVE1450
 1090 CONTINUE                                                          INVE1460
      DO 1110 I=1, N                                                    INVE1470
      L = N+1-I                                                         INVE1480
      IF( INDXR(L)  .EQ.  INDXC(L) )  GO TO 1110                        INVE1490
      IROW = INDXR(L)                                                   INVE1500
      ICOL = INDXC(L)                                                   INVE1510
      DO 1100 K=1, N                                                    INVE1520
      T = A(K,IROW)                                                     INVE1530
      A(K,IROW) = A(K,ICOL)                                             INVE1540
 1100 A(K,ICOL) = T                                                     INVE1550
 1110 CONTINUE                                                          INVE1560
C                                                                       INVE1570
C                                                                       INVE1580
C     SUCCESSFUL SOLUTION                                               INVE1590
C                                                                       INVE1600
C ------------------                                                    INVE1610
      RETURN                                                            INVE1620
C ------------------                                                    INVE1630
C                                                                       INVE1640
 1120 CONTINUE                                                          INVE1650
C                                                                       INVE1660
C     SINGULAR MATRIX                                                   INVE1670
      WRITE (6,1130)                                                    INVE1680
C                                                                       INVE1690
C ------------------                                                    INVE1700
      GOTON = 2
      RETURN
C ------------------                                                    INVE1720
 1130 FORMAT (29H ERROR THE MATRIX IS SINGULAR)                         INVE1730
      END                                                                      -
CDECK SINVRT
      SUBROUTINE SINVRT (C1, NROWS, N, IR1, IR2, SCALE, DET, NDETXP)    SINV1000
C                                                                       SINV1010
C ..................................................................... SINV1020
C                                                                       SINV1030
C ..................................................................... SINV1040
C                                                                       SINV1050
      REAL C1(NROWS, NROWS)                                             SINV1060
      INTEGER N1(115), N2(115), N3(115)                                 SINV1070
C                                                                       SINV1080
      IR1 = 0                                                           SINV1090
      IR2 =0                                                            SINV1100
      SCALE = 0.0                                                       SINV1110
      DET = 0.0                                                         SINV1120
      NDETXP = 0.0                                                      SINV1130
      CALL INVERT (GOTON,C1,N,N1,N2,N3,NROWS)
      IF (GOTON.EQ.2) GO TO 1000
C                                                                       SINV1150
C ------------------                                                    SINV1160
      RETURN                                                            SINV1170
C ------------------                                                    SINV1180
 1000 CONTINUE                                                          SINV1190
      IR1 =2                                                            SINV1200
```


MCDONNELL DOUGLAS

```
C                                                                       SINV1210
C -------------------                                                   SINV1220
      RETURN                                                            SINV1230
C -------------------                                                   SINV1240
      END                                                                  -
CDECK SIMP
      SUBROUTINE SIMP(R,X,Y,N,IER)                                      SIMP1000
      DIMENSION X(N),Y(N)                                               SIMP1010
      R=0.0                                                             SIMP1020
      IF(N.GT.1) GO TO 1000                                             SIMP1030
      IER=2                                                             SIMP1040
      RETURN                                                            SIMP1050
 1000 IF(X(1).EQ.X(2)) GO TO 1110                                       SIMP1060
      NM1=N-1                                                           SIMP1070
      IF(N.EQ.2) GO TO 1120                                             SIMP1080
      IF(X(1).LT.X(2)) GO TO 1020                                       SIMP1090
C  TEST FOR X TO BE MONOTONICALLY DECREASING                            SIMP1100
      DO 1010 I=2,NM1                                                   SIMP1110
      IF(X(I+1).GE.X(I)) GO TO 1110                                     SIMP1120
 1010 CONTINUE                                                          SIMP1130
      GO TO 1040                                                        SIMP1140
C  TEST FOP X TO BE MONOTONICALLY INCREASING                            SIMP1150
 1020 DO 1030 I=2,NM1                                                   SIMP1160
      IF(X(I+1).LE.X(I)) GO TO 1110                                     SIMP1170
 1030 CONTINUE                                                          SIMP1180
 1040 NM2=N-2                                                           SIMP1190
      IF(MOD(N,2).EQ.0) GO TO 1130                                      SIMP1200
      P=0.0                                                             SIMP1210
      N1=1                                                              SIMP1220
 1050 S1=X(N1+1)-X(N1)                                                  SIMP1230
      S2=X(N1+2)-X(N1+1)                                                SIMP1240
      S3=X(NM1)-X(NM2)                                                  SIMP1250
      S4=X(N)-X(NM1)                                                    SIMP1260
      R=(2.*S1**2+S1*S2-S2**2)/S1*Y(N1)+(2.*S4**2+S3*S4-S3**2)/S4*Y(N)  SIMP1270
      N1=N1+1                                                           SIMP1280
      DO 1060 I=N1,NM1,2                                                SIMP1290
      S1=X(I)-X(I-1)                                                    SIMP1300
      S2=X(I+1)-X(I)                                                    SIMP1310
 1060 R=R+(S1+S2)**3/(S1*S2)*Y(I)                                       SIMP1320
      IF(N.LT.5) GO TO 1080                                             SIMP1330
      N1=N1+1                                                           SIMP1340
      DO 1070 I=N1,NM2,2                                                SIMP1350
      S1=X(I-1)-X(I-2)                                                  SIMP1360
      S2=X(I)-X(I-1)                                                    SIMP1370
      S3=X(I+1)-X(I)                                                    SIMP1380
      S4=X(I+2)-X(I+1)                                                  SIMP1390
 1070 R=R+((2.*S2**2+S1*S2-S1**2)/S2+(2.*S3**2+S3*S4-S4**2)/S3)*Y(I)    SIMP1400
 1080 R=R/6.+P                                                          SIMP1410
 1090 K = LEGVAR(R)
      IF (K.NE.0) GO TO 1100
      IF(K.EQ.1) GO TO 1100                                             SIMP1430
      IER = 1                                                           SIMP1440
      RETURN                                                            SIMP1450
 1100 IER=3                                                             SIMP1460
      RETURN                                                            SIMP1470
 1110 IER=4                                                             SIMP1480
      RETURN                                                            SIMP1490
C  TRAPEZOIDAL RULE FOR N=2                                             SIMP1500
 1120 R=(X(2)-X(1))*(Y(1)+Y(2))/2.0                                     SIMP1510
      GO TO 1090                                                        SIMP1520
C  FIT POLYNOMIAL THRU FIRST 3 POINTS AND INTEGRATE FROM X(1) TO X(2).  SIMP1530
 1130 S1=X(2)-X(1)                                                      SIMP1540
      S2=X(3)-X(1)                                                      SIMP1550
      S3=Y(2)-Y(1)                                                      SIMP1560
      S4=Y(3)-Y(1)                                                      SIMP1570
```

```
      P=S1/6.*(2.*S3+6.*Y(1)+(S2**2*S3-S1**2*S4)/(S2*(S2-S1)))          SIMP1580
      N1=2                                                              SIMP1590
      GO TO 1050                                                        SIMP1600
      END                                                                      -
CDECK INVW
      OVERLAY(WANG, 7, 1)
      PROGRAM INVW
C                                                                       INVW1010
C ..................................................................... INVW1020
C     FOR WING ONLY CASE, STORES MATRIX (A) AND (A) INVERSE ON TAPE     INVW1030
C     MAXIMUM SIZE MATRIX INVERSION =110                                INVW1040
C ..................................................................... INVW1050
C                                                                       INVW1060
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1              NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                       INVW1090
      DIMENSION AWW(115,115)                                            INVW1100
C                                                                       INVW1110
      MDEMEN=115                                                        INVW1120
C                                                                       INVW1130
C     READ (A) MATRIX INTO CORE, WRITE ON TAPE                          INVW1140
      CALL FSF(1,NTAPEA,IRR)                                            INVW1150
      DO 1000 J=1,NWING                                                 INVW1160
      READ (NTAPEA) (AWW(I,J),I=1,NWING)                                INVW1170
      WRITE (NTAPEE) (AWW(I,J),I=1,NWING)                               INVW1180
 1000 CONTINUE                                                          INVW1190
      CALL FOR EOF (NTAPEE)                                             INVW1200
      REWIND NTAPEA                                                     INVW1210
C                                                                       INVW1220
C     INVERT MATRIX (A)                                                 INVW1230
      CALL SINVRT(AWW,MDEMEN,NWING,IRR1,IRR2,SCALE,DET,NDETXP)          INVW1240
      IF (IRR1)1010,1020,1010                                           INVW1250
 1010 WRITE (NTAPEO,1040) IRR1,IRR2,SCALE                               INVW1260
      REWIND NTAPEE                                                     INVW1270
C                                                                       INVW1280
C *****************                                                     INVW1290
      STOP                                                              INVW1300
C *****************                                                     INVW1310
C                                                                       INVW1320
 1020 CONTINUE                                                          INVW1330
      DO 1030 J=1,NWING                                                 INVW1340
      WRITE (NTAPEE) (AWW(I,J),I=1,NWING)                               INVW1350
 1030 CONTINUE                                                          INVW1360
      CALL FOR EOF (NTAPEE)                                             INVW1370
      REWIND NTAPEE                                                     INVW1380
C                                                                       INVW1390
C                                                                       INVW1400
C ------------------                                                    INVW1410
C ------------------                                                    INVW1430
 1040 FORMAT(1H1,38HERROR IN INVERSION OF WING ONLY MATRIX              INVW1440
     1,5X,6HIRR1 =,I3,5X,6HIRR2 =,I3,5X,7HSCALE =,E12.6)                INVW1450
      END                                                                      -
CDECK INVBB
      OVERLAY(WANG, 7, 2)
      PROGRAM INVBB
C                                                                       INVB1010
C ..................................................................... INVB1020
C     FOR WING BODY CASE, PREPARES AERODYNAMIC INFLUENCE MATRIX FOR     INVB1030
C     MATRIX REDUCTION                                                  INVB1040
C     MAXIMUM SIZE MATRIX INVERSION =110                                INVB1050
C ..................................................................... INVB1060
C                                                                       INVB1070
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1              NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                       INVB1100
```

```
      DIMENSION ABB(115,115),AWB(110)                                   INVB1110
C                                                                       INVB1120
C                                                                       INVB1130
      MDEMEN=115                                                        INVB1140
C                                                                       INVB1150
C     READ AERODYNAMIC INFLUENCE MATRIX (INFLUENCE ON BODY DUE TO BODY  INVB1160
C     AND INFLUENCE ON WING DUE TO BODY) INTO CORE                      INVB1170
C                                                                       INVB1180
      CALL FSF(1,NTAPEA,IRR)                                            INVB1190
      DO 1000 J=1,NBODY                                                 INVB1200
      READ (NTAPEA) (ABB(I,J),I=1,NBODY),(AWB(I),I=1,NWING)
      IF ( EOF(NTAPEA) ) 1040, 990
  990 CONTINUE
C                                                                       INVB1220
      WRITE (NTAPED) (AWB(I),I=1,NWING)                                 INVB1230
 1000 CONTINUE                                                          INVB1240
      CALL FOR EOF (NTAPED)                                             INVB1250
C                                                                       INVB1260
C     INVERT AERODYNAMIC INFLUENCE MATRIX (INFLUENCE ON BODY DUE TO BODYINVB1270
      CALL SINVRT(ABB,MDEMEN,NBODY,IRR1,IRR2,SCALE,DET,NDETXP)          INVB1280
      IF (IRR1)1010,1020,1010                                           INVB1290
 1010 CONTINUE                                                          INVB1300
      WRITE (NTAPEO,1050) IRR1,IRR2,SCALE                               INVB1310
      REWIND NTAPEA                                                     INVB1320
C                                                                       INVB1330
C ******************                                                    INVB1340
      STOP                                                              INVB1350
C ******************                                                    INVB1360
C                                                                       INVB1370
 1020 CONTINUE                                                          INVB1380
      DO 1030 J=1,NBODY                                                 INVB1390
      WRITE (NTAPED) (ABB(I,J),I=1,NBODY)                               INVB1400
 1030 CONTINUE                                                          INVB1410
      CALL FOR EOF (NTAPED)                                             INVB1420
      REWIND NTAPED                                                     INVB1430
C                                                                       INVB1440
C ------------------                                                    INVB1450
C          PROGRAM  EXITS
      GO TO 1200
C
C ------------------                                                    INVB1470
C                                                                       INVB1480
C     END OF FILE ON NTAPEA                                             INVB1490
 1040 WRITE (NTAPEO,1060) NTAPEA,J,NBODY,NWING                          INVB1500
C *****************                                                     INVB1510
      STOP                                                              INVB1520
C ******************                                                    INVB1530
 1200 CONTINUE
C
C
 1050 FORMAT(1H1,33HERROR IN INVERSION OF BODY MATRIX,5X,               INVB1540
     16HIRR1 =,I3,5X,6HIRR2 =,I3,5X,7HSCALE =,E12.6)                    INVB1550
 1060 FORMAT (1H0,33H****END OF FILE DETECTED ON UNIT  ,I2,3H J= ,I5,   INVB1560
     1     8H  NBODY= ,I3,8H  NWING= ,I3)                               INVB1570
      END                                                                      -
CDECK INVRW
      OVERLAY(WANG, 7, 3)
      PROGRAM INVRW
C                                                                       INVR1010
C ..................................................................... INVR1020
C     INVERT REDUCED MATRIX                                             INVR1030
C     MAXIMUM SIZE MATRIX INVERSION =110                                INVR1040
C ..................................................................... INVR1050
C                                                                       INVR1060
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
```

```
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                         INVR1090
      DIMENSION ARWW(115,115)                                             INVR1100
C                                                                         INVR1110
      MDEMEN=115                                                          INVR1120
      DO 1000 J=1,NWING                                                   INVR1130
      READ (NTAPEE) (ARWW(I,J),I=1,NWING)                                 INVR1140
 1000 CONTINUE                                                            INVR1150
      CALL FOR EOF (NTAPEE)                                               INVR1160
C                                                                         INVR1170
C     MATRIX INVERSION                                                    INVR1180
      CALL SINVRT(ARWW,MDEMEN,NWING,IRR1,IRR2,SCALE,DET,NDETXP)           INVR1190
      IF (IRR1)1010,1020,1010                                             INVR1200
 1010 CONTINUE                                                            INVR1210
      WRITE (NTAPEO,1040) IRR1,IRR2,SCALE                                 INVR1220
      REWIND NTAPEE                                                       INVR1230
C                                                                         INVR1240
C ******************                                                      INVR1250
      STOP                                                                INVR1260
C ******************                                                      INVR1270
C                                                                         INVR1280
 1020 CONTINUE                                                            INVR1290
      DO 1030 J=1,NWING                                                   INVR1300
      WRITE (NTAPEE) (ARWW(I,J),I=1,NWING)                                INVR1310
 1030 CONTINUE                                                            INVR1320
      CALL FOR EOF (NTAPEE)                                               INVR1330
      REWIND NTAPEE                                                       INVR1340
C                                                                         INVR1350
C                                                                         INVR1360
C ------------------                                                      INVR1370
C ------------------                                                      INVR1390
 1040 FORMAT(1H1,36HERROR IN INVERSION OF REDUCED MATRIX,5X,              INVR1400
     16HIRR1 =,I3,5X,6HIRR2 =,I3,5X,7HSCALE =,E12.6)                      INVR1410
      END                                                                 -
CDECK MDMATE
      OVERLAY(WANG, 7, 4)
         PROGRAM MDMATE
C                                                                         MDMA1010
C ....................................................................... MDMA1020
C     MAXIMUM SIZE MATRIX INVERSION =112                                  MDMA1030
C     MUST BE 2 PLUS MAXIMUM NUMBER OF WING PANELS(110)                   MDMA1040
C ....................................................................... MDMA1050
C                                                                         MDMA1060
C                                                                         MDMA1070
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
C                                                                         MDMA1100
      DIMENSION WW(115,115),XBAR(210),AREA(210)                           MDMA1110
C                                                                         MDMA1120
      NN=NWING+1                                                          MDMA1130
      NNN=NWING+2                                                         MDMA1140
C
C        DRAG MINIMIZATION OPTION ELIMINATED BY SETTING DMIN = 0.0 HERE
C        CAN BE RETORED BY PLACE THE STATEMENT " READ (LI,1020) DMIN  "
C        HERE AND REMOVE DMIN = 0.0.
C
      DMIN = 0.0
C
      IF(DMIN)1000,1000,1020                                              MDMA1160
 1000 CALL FSF(1,NTAPEA,IRR)                                              MDMA1170
      DO 1010 I=1,NNN                                                     MDMA1180
      DO 1010 J=1,NNN                                                     MDMA1190
 1010 WW(I,J)=0.                                                          MDMA1200
      GO TO 1130                                                          MDMA1210
 1020 MDEMEN=115                                                          MDMA1220
```

```
      READ (NTAPEC) NBODY,NWING,XMACH,SYM,KACE                          MDMA1230
      NPANEL=NBODY+NWING                                                MDMA1240
      READ (NTAPEC) (J, XBAR(I), DM1, DM2, DM3, DM4, DM5
     1,AREA(I),DUMMY6,DUMMY7,DUMMY8,I=1,NPANEL)                         MDMA1260
      REWIND NTAPEC                                                     MDMA1270
      DO 1030 J=1,NWING                                                 MDMA1280
      READ (NTAPEE) (WW(I,J),I=1,NWING)                                 MDMA1290
 1030 CONTINUE                                                          MDMA1300
      REWIND NTAPEE                                                     MDMA1310
      DO 1040 I=1,NWING                                                 MDMA1320
      K=I                                                               MDMA1330
      II=I+NBODY                                                        MDMA1340
      DO 1040 J=1,K                                                     MDMA1350
      JJ=J+NBODY                                                        MDMA1360
      WW(I,J)=-WW(I,J)*AREA(II)-WW(J,I)*AREA(JJ)                        MDMA1370
      WW(J,I)=WW(I,J)                                                   MDMA1380
 1040 CONTINUE                                                          MDMA1390
      DO 1050 J=1,NWING                                                 MDMA1400
      JJ=J+NBODY                                                        MDMA1410
      WW(J,NN)=-AREA(JJ)                                                MDMA1420
      WW(NN,J)=-AREA(JJ)                                                MDMA1430
      XA=-XBAR(JJ)*AREA(JJ)                                             MDMA1440
      WW(J,NNN)=XA                                                      MDMA1450
      WW(NNN,J)=XA                                                      MDMA1460
 1050 CONTINUE                                                          MDMA1470
      WW(NN,NN)=0.0                                                     MDMA1480
      WW(NN,NNN)=0.0                                                    MDMA1490
      WW(NNN,NN)=0.0                                                    MDMA1500
      WW(NNN,NNN)=0.0                                                   MDMA1510
      CALL FSF(1,NTAPEA,IRR)                                            MDMA1520
      DO 1060 J=1,NNN                                                   MDMA1530
 1060 WRITE (NTAPEA)(WW(I,J),I=1,NNN)                                   MDMA1540
      CALL FOR EOF (NTAPEA)                                             MDMA1550
C                                                                       MDMA1560
C     INVERT DRAG MINIMIZATION MATRIX CONSTRAINED FOR CL                MDMA1570
      CALL SINVRT (WW,MDEMEN,NN,IRR1,IRR2,SCALE,DET,NDETXP)             MDMA1580
      IF (IRR1)1070,1080,1070                                           MDMA1590
 1070 CONTINUE                                                          MDMA1600
      WRITE (NTAPEO,1170) IRR1,IRR2,SCALE                               MDMA1610
      REWIND NTAPEA                                                     MDMA1620
C                                                                       MDMA1630
C ******************                                                    MDMA1640
      STOP                                                              MDMA1650
C ******************                                                    MDMA1660
C                                                                       MDMA1670
 1080 CONTINUE                                                          MDMA1680
      DO 1090 J=1,NN                                                    MDMA1690
 1090 WRITE (NTAPEA)(WW(I,J),I=1,NN)                                    MDMA1700
      CALL FOR EOF (NTAPEA)                                             MDMA1710
      REWIND NTAPEA                                                     MDMA1720
      CALL FSF(1,NTAPEA,IRR)                                            MDMA1730
      DO 1100 J=1,NNN                                                   MDMA1740
 1100 READ (NTAPEA)(WW(I,J),I=1,NNN)                                    MDMA1750
      CALL FSF(2,NTAPEA,IRR)                                            MDMA1760
C                                                                       MDMA1770
C     INVERT DRAG MINIMIZATION MATRIX CONSTRAINED FOR CL AND CM         MDMA1780
      CALL SINVRT (WW,MDEMEN,NNN,IRR1,IRR2,SCALE,DET,NDETXP)            MDMA1790
      IF (IRR1)1070,1110,1070                                           MDMA1800
 1110 CONTINUE                                                          MDMA1810
      DO 1120 J=1,NNN                                                   MDMA1820
 1120 WRITE (NTAPEA)(WW(I,J),I=1,NNN)                                   MDMA1830
      CALL FOR EOF (NTAPEA)                                             MDMA1840
      REWIND NTAPEA                                                     MDMA1850
C                                                                       MDMA1860
C ------------------                                                    MDMA1870
```

```
C        PROGRAM  EXITS
      GO TO 1200
C
C -------------------                                                   MDMA1890
C                                                                       MDMA1900
 1130 DO 1140 J=1,NNN                                                   MDMA1910
 1140 WRITE (NTAPEA)(WW(I,J),I=1,NNN)                                   MDMA1920
      CALL FOR EOF (NTAPEA)                                             MDMA1930
      DO 1150 J=1,NN                                                    MDMA1940
 1150 WRITE (NTAPEA)(WW(I,J),I=1,NN)                                    MDMA1950
      CALL FOR EOF (NTAPEA)                                             MDMA1960
      GO TO 1110                                                        MDMA1970
C                                                                       MDMA1980
 1200 CONTINUE
C                                                                       MDMA1990
C     FORMATION OF DRAG MINIMIZATION MATRIX                             MDMA2000
 1160 FORMAT(7F10.0)                                                    MDMA2010
 1170 FORMAT(1H1,46HERROR IN INVERSION OF DRAG MINIMIZATION MATRIX      MDMA2020
     1,6HIRR1 =,I3,5X,6HIRR2 =,I3,5X,7HSCALE =,E12.6)                   MDMA2030
      END                                                               -
CDECK WANG75
      OVERLAY(WANG, 7, 5)
      PROGRAM WANG75
C
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON /AVAR/  A(210),ACB(21),ABX(100),AWX(110),AREA(210),ARN(2),A00000090
     1               RWT(20),ALPHA(210),ALPHAB(210),ALPHAS(210),ALPHAT(100000100
     2               10),ALPHAW(110),ALPHAX(110),AWS(110),ABT(100),AN1N(00000110
     3               210,2),ALPHAA,ALPHAD,ARA,ARADEG,ARB,ARW,ARAS,AT,AAA00000120
      COMMON /BODS/ NUMBOD,XNOS(3),YNOS(3),ZNOS(3),NBSTAT(3)            00000130
      COMMON /BVAR/  B(210)                                             00000140
     1               ,BBCL,BBCD,BBCM                                    00000150
      COMMON/CVAR/C(210), CHORD(210),CL(210),CPNN(51,11,2)
     1               ,CLS(210)                                          00000170
     2               ,CDR,CASE,CPCALC,CBAR,CONSNT,CLBAR,CLX,CLM,CDM     00000180
      COMMON /KVAR/  KASE,KONFIG,KPOLAR                                 00000220
      COMMON /NVAR/  NFMT(9),NROW(2)                                    00000230
     1          ,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,NTHETS,
     2          NXLE, NRG, NPOLR, NCLX
      COMMON/RVAR/ R(51), RN(51,2), RATIOX, RFAREA
      COMMON/TVAR/ TITLE(20), THETA(210), THETAB(11),THETAS(11), THETAN
     1          (11,2), TCL(11), TCD(11), TCM(11), THICK,TWIST
      COMMON /UVAR/  UEWT(100),UWBT(110),UWWT(110),UNCL(210,2)          00000330
      COMMON /VVAR/  VBWT(100),VWBT(110),VWWT(110),VNCL(210,2)          00000340
     1               ,VOUT                                              00000350
      COMMON /WVAR/  WBWT(100),WWBT(110),WWWT(110),WNCL(210,2)          00000360
      COMMON/XVAR/ XBAR(210), XC(210), XNI(2), XNXN(2), XNTN(2), XB(51),
     1          XN(51,2), XNNCD(2), XNNCL(2), XNNCM(2), XYZ(3), XCL(11),
     2          XCD(11), XNACEL, XP, XCPBAR
      COMMON /YVAR/  YBAR(210),YC(210),YNI(2)                           00000410
      COMMON/ZVAR/ ZBAR(210), ZC(210), ZNI(2), ZDELTA(51), ZDN(51,2)
     1               ,ZP,ZA                                             00000430
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1          ,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH
      COMMON/EXTRA/ ARAN, DSB1, DSB2, DS1, DS2, NB1, NB2, NS1, NS2,
     1          NTN, NXN, XNOSE
      COMMON/NSLOPE/ DRNDX(51,2), DZNDX(51,2)
      DIMENSION NS1(2), NS2(2), DS1(2), DS2(2)
C
C
      IF ( KONFIG  .EQ. 1 ) GO TO 50
      CALL SUBBOD(NBODYS, NTHETB, 0, XNOSE, 0.0, 0.0, CPCALC, VOUT,ARA,
```

```
     1         XP, ZP, RFAREA, XB, R, THETB, ZDELTA, XC, YC, ZC, THETA,
     2         UWBT, VWBT, WWBT, ALPHAX, CPBB, BBCL, BBCD, BBCM, NB1,
     3         NB2, DSB1, DSB2)
C
   50 CONTINUE
      IF ( XNACEL ) 100, 500, 100
  100 CONTINUE
C
      DO 200 N=1, NACEL
C
      DO 150 J=1,NXN
      DRDX(J) = DRNDX(J,N)
      DZDX(J) = DZNDX(J,N)
  150 CONTINUE
C
      ARAN = ARA + ARN(N) / CDR
      CALL SUBBOD( NXN, NTN, NACEL, XNI(N), YNI(N), ZNI(N), CPCALC, VOUT
     1         , ARAN, XP, ZP, RFAREA, XN(1,N), RN(1,N), THETAN(1,N),
     2         ZDN(1,N), XC, YC, ZC, THETA, UNCL(1,N), VNCL(1,N),
     3         WNCL(1,N), AN1N(1,N), CPNN(1,1,N), XNNCL(N), XNNCD(N),
     4         XNNCM(N), NS1(N), NS2(N), DS1(N), DS2(N) )
  200 CONTINUE
C
  500 CONTINUE
C
      END
CDECK TRAP
      SUBROUTINE TRAP(XT,YT,ZT,NT)                                      TRAP1000
      DIMENSION XT(1),YT(1)                                             TRAP1010
      SUM=0.                                                            TRAP1020
      DO 1000 I=2,NT                                                    TRAP1030
 1000 SUM=SUM+.5*(XT(I)-XT(I-1))*(YT(I)+YT(I-1))                        TRAP1040
      ZT=SUM                                                            TRAP1050
      RETURN                                                            TRAP1060
      END                                                                      -
CDECK SUBBOD
      SUBROUTINE SUBBOD(NBS,NT,NSTOR,XN,YN,ZN,CPC,VEL,ARA,XP,ZP,RFAREA, 00000010
     1XB,R,THETB,ZCAM,XC,YC,ZC,TA,U,V,W,EN,CPB,BCL,BCD,BCM,             00000020
     2NS1,NS2,DS1,DS2)                                                  00000030
C     COMPUTES SINGULARITY STRENGTHS,VELOCITY COMPONENTS AND PRESSURES  00000040
C ..................................................................... 00000050
C                                                                       00000060
C ..................................................................... 00000070
C                                                                       00000080
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1         NTAPEO,NBOD ,NW    ,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /.BODYSP/ DRDX(51), DZDX(51)
C                                                                       00000110
      DIMENSION XB(1),R(1),THETB(1),ZCAM(1),XC(1),YC(1),                00000120
     1         ZC(1), TA(1), U(1), V(1), W(1), EN(1), CPB(51,1), QC(51),
     2         THETD(11), RR(210), SY(2), RB(51), UB(51,11), VB(51,11),
     3         WB(51,11), VD(51,51), UC(51), VC(51),RB2(51)
     4,X(51),ARC(51),Q(51),ZB(51)                                       00000160
      DIMENSION A(52),B(52),C(52),XX(52),YA(11),YX(11)
      DIMENSION  CP(11),G(52)                                           00000180
C                                                                       00000190
      EQUIVALENCE ( UC(1), RR(1)), ( VC(1), RR(52))
C                                                                       00000210
C                                                                       00000220
C     CALCULATION OF SINGULARITY STRENGTHS                              00000230
      NP=NW+NBOD                                                        00000240
      NB=NBS-1                                                          00000250
      KA=0                                                              00000260
      PI=3.1415926                                                      00000270
      B2=1.-EM*EM                                                       00000280
```

```
      DO 1000 N=1,NB
      Q(N)=0.
 1000 QC(N)=0.
      X1=XB(1)
      DO 1010 N=1,NBS
 1010 XB(N)=XB(N)-X1
      DO 1020 N=1,NT
 1020 THETB(N)=THETB(N)/57.2957795
      DO 1040 N=1,NB
      ARC(N)=ARA-DZDX(N)
      RB(N)=(R(N+1)+R(N))/2.
      RB2(N)=RB(N)**2
      ZB(N)=(ZCAM(N+1)+ZCAM(N))/2.
      X(N)=(XB(N+1)+XB(N))/2.
      IF(ARC(N))1030,1040,1030
 1030 KA=1
 1040 CONTINUE
      DO 1060 N=1,NB
      DO 1060 M=1,NB
      EX=X(N)-XB(M)
      EL=XB(NBS)-XB(M)
      D1=SQRT((EX-EL)**2+B2*RB2(N))
      D2=SQRT(EX**2+B2*RB2(N))
      US=1./D1-1./D2
      VS=B2*RB(N)*    (1./(D1*(EX-EL+D1))-1./(D2*(EX+D2)))
 1060 VD(N,M)=VS-DRDX(N)*US
      GO TO 1090
 1070 DO 1080 N=1,NB
      DO 1080 M=1,NB
      EX=X(N)-XB(M)
      EL=XB(NBS)-XB(M)
      D1=((EX-EL)**2+B2*RB2(N))**1.5
      D2=(EX**2+B2*RB2(N))**1.5
      UR=B2*RB(N)*(1./D2-1./D1)
      VR=((EX-EL)*((EX-EL)**2+2.*B2*RB2(N))/D1-EX*(EX**2+2.*B2*RB2(N))
     1/D2)/RB2(N)
 1080 VD(N,M)=VR-DRDX(N)*UR
 1090 CALL SINVRT(VD,51,NB,I1,I2,SC,DET,NDETXP)
      IF(I1)1100,1110,1100
 1100 WRITE (NTAPEO,1590) I1,I2,SC
 1110 IF(KA)1150,1120,1120
 1120 DO 1130 N=1,NB
      Q(N)=0.
      DO 1130 M=1,NB
 1130 Q(N)=Q(N)+VD(N,M)*DRDX(M)
      IF(KA)1170,1170,1140
 1140 KA=-1
      GO TO 1070
 1150 DO 1160 N=1,NB
      QC(N)=0.
      DO 1160 M=1,NB
 1160 QC(N)=QC(N)-VD(N,M)*ARC(M)
 1170 CONTINUE
C
C     VELOCITY COMPONENTS ON THE BODY
C
      DO 1210 N=1,NB
C
      U(N)=0.
      UC(N)=0.
      V(N)=0.
      VC(N)=0.
      W(N)=0.
      DO 1190 M=1,NB
      EX=X(N)-XB(M)
```

```
      EL=XB(NBS)-XB(M)
      XL=EX-EL
      D1=SQRT(XL**2+B2*RB2(N))
      D2=SQRT(EX**2+B2*RB2(N))
      D3=D1**3.
      D4=D2**3.
      US=1./D1-1./D2
      VS=B2*RB(N)*(1./(D1*(XL+D1))-1./(D2*(EX+D2)))
      U(N)=U(N)+US*Q(M)
      V(N)=V(N)+VS*Q(M)
      IF(KA)1180,1190,1180
 1180 UR=B2*RB(N)*(1./D4-1./D3)
      VR=(XL*(XL**2+2.*B2*RB2(N))/D3-EX*(EX**2+2.*B2*RB2(N))/D4)/RB2(N)
      WR=(XL/D1-EX/D2)/RB2(N)
      UC(N)=UC(N)+UR*QC(M)
      VC(N)=VC(N)+VR*QC(M)
      W(N)=W(N)+WR*QC(M)
 1190 CONTINUE
      DO 1200 L=1,NT
      UB(N,L)=U(N)+UC(N)*COS(THETB(L))
      VB(N,L)=V(N)+VC(N)*COS(THETB(L))
 1200 WB(N,L)=W(N)*SIN(THETB(L))
C
 1210 CONTINUE
C
C     VELOCITY COMPONENTS ON THE WING
      IF(NW)1300,1300,1220
 1220 IF(NSTOR)1230,1240,1230
 1230 NS=1
      JM=2
      SY(1)=1.
      SY(2)=-1.
      GO TO 1250
 1240 NS=NBOD+1
      JM=1
      SY(1)=0.
 1250 CONTINUE
      L=0
C
      DO 1290 N=NS,NP
C
      L=L+1
      U(L)=0.
      V(L)=0.
      W(L)=0.
      EN(L)=0.
      DO 1290 J=1,JM
      DYC=YC(N)-SY(J)*YN
      DZC=ZC(N)-ZN
      IF(DYC.EQ.0..AND.DZC.EQ.0.) GO TO 1260
      THETN=ATAN2(DYC,DZC)
      GO TO 1270
 1260 THETN=0.
 1270 COSTN=COS(THETN)
      SINTN=SIN(THETN)
      R2=(YC(N)-SY(J)*YN)**2+(ZC(N)-ZN)**2
      RR(L)=SQRT(R2)
      BR2=B2*R2
      VR=0.
      VT=0.
      DO 1280 M=1,NB
      EX=XC(N)-XN-XB(M)
      EL=XB(NBS)-XB(M)
      XL=EX-EL
      D1=SQRT(XL**2+BR2)
```

```
      D2=SQRT(EX**2+BR2)
      D3=D1**3
      D4=D2**3
      U(L)=U(L)+(1./D1-1./D2)*Q(M)+B2*RR(L)*(1./D4-1./D3)*COSTN*QC(M)
      VR=VR+B2*RR(L)*(1./(D1*(XL+D1))-1./(D2*(EX+D2)))*Q(M)+(XL*(XL**2+2
     1.*BR2)/D3-EX*(EX**2+2.*BR2)/D4)/R2*QC(M)*COSTN
 1280 VT=VT+(XL/D1-EX/D2)/R2*QC(M)*SINTN
      THETS = THETN + TA(N)
      V(L)=VR*SINTN+VT*COSTN+V(L)
      W(L)=VR*COSTN-VT*SINTN+W(L)
      EN(L)=EN(L)+VR*COS(THETS)-VT*SIN(THETS)
C
 1290 CONTINUE
C
C     CALCULATE PRESSURES, FORCES, AND MOMENTS
C
 1300 ARM = ARA
      SINARM = SIN(ARM)
      COSARM = COS(ARM)
      IF (EM .EQ. 0.) GO TO 1310
      Q000 =  1.42857/EM**2
      Q001 =   .2*EM**2
 1310 DO 1360 N = 1, NT
      COST = COS(THETB(N))
      SINT = SIN(THETB(N))
      DO 1360 M = 1, NB
      VPM   = WB(M,N) * COST + VB(M,N) * SINT
      WPM   = VB(M,N) * COST - WB(M,N) * SINT
      UWPM =  UB(M,N) * COSARM  +  WPM *  SINARM
      UWIND = 1. + UWPM
      VWIND = VPM
      WWIND = WPM * COSARM - UB(M,N) * SINARM
      IF (CPC - 1.) 1320,1330,1340
 1320 CPB(M,N) = -2. * UWPM
      GO TO 1360
 1330 CPB(M,N) = -2.*UWPM -B2*UWPM**2 - VWIND**2 - WWIND**2
      GO TO 1360
 1340 Q2 = UWIND**2+VWIND**2+WWIND**2
      IF (EM .EQ. 0.) GO TO  1350
      CPB(M,N) = Q000* ((1. + Q001*(1.-Q2)) **3.5 - 1.)
      GO TO 1360
 1350 CPB(M,N) = 1. - Q2
 1360 CONTINUE
C
      IF(EM)1390,1380,1370
 1370 CPSTAG = Q000*((1.+Q001)**3.5-1.)
      GO TO 1390
 1380 CPSTAG=1.
 1390 CONTINUE
      BCL=0.
      BCD=0.
      BCM=0.
      IF(NS1)1410,1410,1400
 1400 IF(NS2)1410,1410,1420
 1410 NS1=1
      NS2=NBS
 1420 NBL=NS2+1-NS1
      NBT=NBL+1
      M1=NS2-1
      MM = 1
      DO 1440 M = NS1,M1
      MM = MM + 1
      RM = RB(M)
      DO 1430  N = 1,NT
      COST = COS(THETB(N))
```

```
      CPBB = CPB(M,N)
      CP(N) = CPBB
      YA(N) =  CPBB * COST
 1430 YX(N) = YA(N) * COST
      CALL SIMP (GM,THETB,CP,NT,IRR)
      CALL TRAP (THETB,YA,AM,NT)
      CALL TRAP (THETB, YX, BM, NT)
      IF (IRR .NE. 1) GO TO 1580
      A(MM) = AM * RM
      B(MM) = BM * RM
      G(MM) = GM * RM
 1440 CONTINUE
      NN = NS1 - 1
      DO 1450 N = 2,NBL
      NN = NN + 1
      C(N) = DZDX(NN) * A(N) + DRDX(NN) * G(N)
 1450 XX(N) = X(NN)
      XX(1) = XB(NS1)
      XX(NBT) = XB(NS2)
      IF (VEL.GT.0.) WRITE (6,1710)
      IF (VEL.GT.0.) WRITE (6,1720)
      IF (VEL.GT.0.) WRITE (6,1700) (XX(N),N= 1,NBT)
      C(1) = .5 * CPSTAG * DS1
      C(NBT) = .5 * CPSTAG * DS2
      IF (R(NBS) .NE. 0.)  C(NBT) = 0.
      A(1) = ZCAM(1)*C(1)
      A(NBT) = ZCAM(NBS)*C(NBT)
      IF (VEL.GT.0.) WRITE (6,1730)
      IF (VEL.GT.0.) WRITE (6,1700) (A(N), N = 1,NBT)
      IF (VEL.GT.0.) WRITE (6,1740)
      IF (VEL.GT.0.) WRITE (6,1700) (C(N), N = 1,NBT)
      CALL SIMP (CN,XX,A,NBT,IRR)
      CN=-CN
      IF (IRR .NE. 1) GO TO 1580
      CALL SIMP (CX,XX,C,NBT,IRR)
      IF (IRR .NE. 1) GO TO 1580
      NN = NS1 - 1
      DO 1460 N = 2,NBL
      NN = NN + 1
 1460 A(N) = A(N)*(X(NN)+XN-XP+RB(NN)*DRDX(NN))+ZB(NN)*C(N)+RB(NN)
     * *DZDX(NN)*B(N)
      IF(VEL.GT.0.) WRITE (6,1750)
      IF(VEL.GT.0.) WRITE (6,1700) (A(N), N = 1,NBT)
      CALL SIMP (CM,XX,A,NBT,IRR)
      IF (IRR .NE. 1) GO TO 1580
      BCL=CN*COS(ARA)-CX*SIN(ARA)
      BCD=CX*COS(ARA)+CN*SIN(ARA)
      BCM = CM + CX*(ZN-ZP)
      DO 1480 N=1,NBS
 1480 XB(N)=XB(N)+X1
      DO 1490 N=1,NT
 1490 THETB(N)=THETB(N)*57.2957795
C
C     PRINT OUT VELOCITY COMPONENTS
      IF(VEL)1500,1570,1500
 1500 IF(NSTOR)1510,1520,1510
 1510 WRITE (NTAPEO,1620) NSTOR,NSTOR
      GO TO 1530
 1520 WRITE (NTAPEO,1630)
 1530 CONTINUE
      WRITE(NTAPEO,1660)
      WRITE (NTAPEO,1640) (THETB(M),M=1,NT)
      WRITE(NTAPEO,1650)
      DO 1540 N=1,NB
 1540 WRITE (NTAPEO,1610) X(N),(UB(N,M),M=1,NT)
```

```
      WRITE(NTAPEO,1670)
      WRITE (NTAPEO,1640) (THETB(M),M=1,NT)
      WRITE(NTAPEO,1650)
      DO 1550 N=1,NB
 1550 WRITE (NTAPEO,1610) X(N),(VB(N,M),M=1,NT)
      WRITE(NTAPEO,1680)
      WRITE (NTAPEO,1640) (THETB(M),M=1,NT)
      WRITE(NTAPEO,1650)
      DO 1560 N=1,NB
 1560 WRITE (NTAPEO,1610) X(N),(WB(N,M),M=1,NT)
      WRITE(6,1760)(X(N),ZB(N),RB(N),DRDX(N),DZDX(N),Q(N),QC(N),N=1,NB)
 1570 CONTINUE
      WRITE(NTAPEC) NBS
      WRITE(NTAPEC) (XB(J),R(J),J=1,NBS)
      WRITE(NTAPEC) (Q(J),QC(J),J=1,NBS)
C
C -------------------
      RETURN
C -------------------
C
 1580 WRITE (NTAPEO,1690) IRR
      GO TO 1570
C
C
C     FORMAT STATEMENTS
 1590 FORMAT (1H1,5X,16H ERROR CODE 1 = I5,16H ERROR CODE 2 = I5,
     115H SCALE FACTOR = E16.8)
 1610 FORMAT(1H ,F14.4,F15.5,9F10.5/(F29.5,9F10.5))
 1620 FORMAT (1H1,36H VELOCITY COMPONENTS ON STORE PAIR (,I2,
     121H) DUE TO STORE PAIR (,I2,27H) LINE SOURCES AND DOUBLETS)
 1630 FORMAT(1H1//66H VELOCITY COMPONENTS ON BODY DUE TO BODY LINE SOURC
     1ES AND DOUBLETS)
 1640 FORMAT(1H0,4X,11HTHETA(DEG.),F14.4,9F10.4/(F29.4,9F10.4))
 1650 FORMAT(1H ,9X,1HX)
 1660 FORMAT(1H0,5X,8HAXIAL(U))
 1670 FORMAT(1H0,5X,10HRADIAL(VR))
 1680 FORMAT(1H0,5X,14HTANGENTIAL(VT))
 1690 FORMAT (1H0,18HINTEGRATION ERROR=I2)
 1700 FORMAT(1H F11.5,9F12.5)
 1710 FORMAT(1H0/39H FORCE AND MOMENT INTEGRATION VARIABLES)
 1720 FORMAT(1H0/ 8H X ARRAY)
 1730 FORMAT(1H0/19H NORMAL FORCE ARRAY)
 1740 FORMAT(1H0/18H AXIAL FORCE ARRAY)
 1750 FORMAT(1H0/22H PITCHING MOMENT ARRAY)
 1760 FORMAT(1H1//35H BODY GEOMETRY AND SOURCE STRENGTHS/
     156X,5HFIRST,9X,6HCAMBER,11X,1HQ,14X,2HQC/9X,
     29HX-STATION,7X,6HCAMBER,9X,6HRADIUS,7X,10HDERIVATIVE,9X,6HSLOPES//
     3(1X,7F15.5))
      END
CDECK WANG76
      OVERLAY(WANG, 7, 6)
      PROGRAM WANG76
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON /AVAR/  A(210),ACB(21),ABX(100),AWX(110),AREA(210),ARN(2),A
     1               RWT(20),ALPHA(210),ALPHAB(210),ALPHAS(210),ALPHAT(1
     2               10),ALPHAW(110),ALPHAX(110),AWS(110),ABT(100),AN1N(
     3               210,2),ALPHAA,ALPHAD,ARA,ARADEG,ARB,ARW,ARAS,AT,AAA
      COMMON /BODS/  NUMBOD,XNOS(3),YNOS(3),ZNOS(3),NESTAT(3)
      COMMON /BVAR/  B(210)
     1               ,BBCL,BBCD,BBCM
      COMMON/CVAR/C(210), CHORD(210),CL(210),CPNN(51,11,2)
     1               ,CLS(210)
```


MCDONNELL DOUGLAS

```
     2              ,CDR,CASE,CPCALC,CBAR,CONSNT,CLBAR,CLX,CLM,CDM       00000180
      COMMON /KVAR/  KASE,KONFIG,KPOLAR                                  00000220
      COMMON /NVAR/  NFMT(9),NROW(2)                                     00000230
     1         ,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,NTHETS,
     2         NXLE, NRG, NPOLR, NCLX
      COMMON/RVAR/ R(51), RN(51,2), RATIOX, RFAREA
      COMMON/TVAR/ TITLE(20), THETA(210), THETAB(11),THETAS(11), THETAN
     1         (11,2), TCL(11), TCD(11), TCM(11), THICK,TWIST
      COMMON /UVAR/  UBWT(100),UWBT(110),UWWT(110),UNCL(210,2)           00000330
      COMMON /VVAR/  VBWT(100),VWBT(110),VWWT(110),VNCL(210,2)           00000340
     1              ,VOUT                                                00000350
      COMMON /WVAR/  WBWT(100),WWBT(110),WWWT(110),WNCL(210,2)           00000360
      COMMON/XVAR/ XBAR(210), XC(210), XNI(2), XNXN(2), XNTN(2), XB(51),
     1         XN(51,2), XNNCD(2), XNNCL(2), XNNCM(2), XYZ(3), XCL(11),
     2         XCD(11), XNACEL, XP, XCPBAR
      COMMON /YVAR/  YBAR(210),YC(210),YNI(2)                            00000410
      COMMON/ZVAR/ ZBAR(210), ZC(210), ZNI(2), ZDELTA(51), ZDN(51,2)
     1              ,ZP,ZA                                               00000430
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1         ,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH
     2         , CPB
      COMMON/EXTRA/ ARAN, DSB1, DSB2, DS1, DS2, NB1, NB2, NS1, NS2,
     1         NTN, NXN, XNOSE
      COMMON/NSLOPE/ DRNDX(51,2), DZNDX(51,2)
      DIMENSION NS1(2), NS2(2), DS1(2), DS2(2)
      DIMENSION CPB(110)
C
C
      IF ( KONFIG  .EQ. 1 ) GO TO 50
      CALL KARMOR(NBODYS, NTHETB, 0, XNOSE, 0.0, 0.0, CPCALC, VOUT,ARA,
     1         XP, ZP, RFAREA, XB, R, THETB, ZDELTA, XC, YC, ZC, THETA,
     2         UWBT, VWBT, WWBT, ALPHAX, CPBB, BBCL, BBCD, BBCM, NB1,
     3         NB2, DSB1, DSB2)
C
   50 CONTINUE
      IF ( XNACEL ) 100, 500, 100
  100 CONTINUE
C
      DO 200 N=1, NACEL
C
      DO 150 J=1,NXN
      DRDX(J) = DRNDX(J,N)
      DZDX(J) = DZNDX(J,N)
  150 CONTINUE
C
      ARAN = ARA + ARN(N) / CDR
      CALL KARMOR( NXN, NTN, NACEL, XNI(N), YNI(N), ZNI(N), CPCALC, VOUT
     1         , ARAN, XP, ZP, RFAREA, XN(1,N), RN(1,N), THETAN(1,N),
     2         ZDN(1,N), XC, YC, ZC, THETA, UNCL(1,N), VNCL(1,N),
     3         WNCL(1,N), AN1N(1,N), CPNN(1,1,N), XNNCL(N), XNNCD(N),
     4         XNNCM(N), NS1(N), NS2(N), DS1(N), DS2(N) )
  200 CONTINUE
C
  500 CONTINUE
C
      END
CDECK KARMOR
      SUBROUTINE KARMOR(NBODYS,NTHETA,NACEL,XN,YN,ZN,CPCALC,VOUT,ARA     00000010
     1,XP,ZP,RFAREA,XB,R,THETAB,ZDELTA,XC,YC,ZC,THETA,U,VV,WW,AN1,CPBB   00000020
     2,BBCL,BBCD,BBCM,NS1,NS2,DS1,DS2)                                   00000030
C                                                                        00000040
C ...................................................................... 00000050
C     CALCULATION OF VELOCITY COMPONENTS DUE TO BODY THICKNESS           00000060
C ...................................................................... 00000070
C                                                                        00000080
```

```
C                                                                       00000090
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
C                                                                       00000120
      DIMENSION XB(1),R(1),THETAB(1),ZDELTA(1),XC(1),YC(1),ZC(1)        00000130
     1         ,U(1), VV(1), WW(1), AN1(1), CPBB(51,1), THETA(1)
      DIMENSION  ALPHAC(51), T(51), TC(51), TX(51), UB(51,11), VB(51,11)
     1         , VTB(51,11), SY(2), FD(51), SD(51), NBS(11), COSTB(11),
     2         SINTB(11), CP(11), YA(11), YB(11), A(51), D(51), G(51),
     3         C(51), XBB(51)
C                                                                       00000190
C                                                                       00000200
      EPS = 1.0E-6                                                      00000210
      PI=3.1415926                                                      00000220
      XMACH2=XMACH*XMACH                                                00000230
      BT2=XMACH2-1.                                                     00000240
      BETA=SQRT(BT2)                                                    00000250
      N1=NBODYS-1                                                       00000260
      X1 = XB(1)                                                        00000261
      DO 1000 N = 1, NBODYS                                             00000311
 1000 XB(N) = XB(N) - X1                                                00000312
      DO 1010 J=1,NTHETA                                                00000320
      THETAB(J) = THETAB(J)/57.2957795                                  00000330
      COSTB(J) = COS(THETAB(J))                                         00000340
 1010 SINTB(J) = SIN(THETAB(J))                                         00000350
C                                                                       00000360
C     COMPUTE FIRST AND SECOND DERIVATIVES                              00000370
      DO 1015 N = 1, NBODYS                                             00000372
 1015 ALPHAC(N) = ARA - DZDX(N)                                         00000373
      DO 115 I=1,N1
      FD(I) = DRDX(I)
  115 CONTINUE
      FD2 = FD(1) * FD(1)                                               00000390
      BTAN = BETA * FD(1)                                               00000400
      IF (BTAN .LT. 1.0) GO TO 1020                                     00000410
      WRITE (6,1580)                                                    00000420
C                                                                       00000430
C ******************                                                    00000440
      STOP                                                              00000450
C ******************                                                    00000460
C                                                                       00000470
 1020 SBT2 = SQRT (1.-BT2*FD2)                                          00000480
      SLOG = ALOG((1.+SBT2)/BTAN)*FD2                                   00000490
C                                                                       00000500
C     CALCULATE SOURCE STRENGTHS                                        00000510
C                                                                       00000520
      DO 1150 I=1,N1                                                    00000530
C                                                                       00000540
      M=I+1                                                             00000550
      UBSUM=0.                                                          00000560
      VBSUM=0.                                                          00000570
      UBDSUM=0.                                                         00000580
      VBDSUM=0.                                                         00000590
      VTDSUM=0.                                                         00000600
      PHIS = 0.                                                         00000610
      PHID=0.                                                           00000620
      RDRDX = FD(M)*R(M)                                                00000630
      RM=R(M)                                                           00000640
      RM2=RM*RM                                                         00000650
      RDRDM=RDRDX                                                       00000660
      RS=RM2*ALPHAC(I)                                                  00000670
      TRM1 = XB(M)-XB(1)                                                00000680
      BRM=BETA*RM                                                       00000690
      BRM2 = BRM*BRM                                                    00000700
```

```
      RABS1 = SQRT(TRM1*TRM1-BRM2)
      IF(BRM.EQ.0.)GO TO 1030
      UBS1 = -ALOG ((TRM1 + RABS1)/BRM)
 1030 RRABD1 = -.5*(TRM1*RABS1-BRM2*UBS1)
      RUBD1 = RABS1
      RVTBD1 = -.5*(TRM1*RABS1 + BRM2*UBS1)
      RN1 = RABS1-RDRDX*UBS1
      RM1 = RRABD1-RDRDX*RUBD1
      IF(I.NE.1)GO TO 1050
C
C     IF I EQUALS 1, DO TANGENT CONE SOLUTION
      T11 = FD2/(SBT2 * SLOG)
      TC11 = FD2*ALPHAC(1)/((FD2+.5)*SBT2 +.5*BT2*SLOG)
      DO 1040 K=1,NTHETA
      UB(1,K) = - SLOG*T11 /FD2
      VB(1,K) = SBT2*T11/FD(1)
      VTB(1,K) = 0.
      IF(RM.EQ.0.) GO TO 1040
      UB(1,K)=UB(1,K)+COSTB(K)*SBT2/FD(1)*TC11
      VB(1,K)=VB(1,K)-.5*COSTB(K)*(SBT2+BT2*SLOG)/FD2*TC11
      VTB(1,K)=-.5*SINTB(K)*(SBT2-BT2*SLOG)/FD2*TC11
 1040 CONTINUE
C
 1050 CONTINUE
      IF (RM .EQ. 0) GO TO 1060
       Q001 = RUBD1*TC11/RM
       Q002 = RABS1*T11/RM
       Q003 = RRABD1*TC11/RM2
       Q004 = RVTBD1*TC11/RM2
 1060 DO 1130 J=1,I
      TRMJ = XB(M)-XB(J) + BETA*R(J)
      TRMJ2 = TRMJ*TRMJ
      IF(BRM.EQ.0.) GO TO 1100
      F1=SQRT(TRMJ2-BRM2)
      F2 = ALOG((TRMJ+F1)/BRM)
      RABS2 = TRMJ*F1-BRM2*F2
      UBS2 = 2.*(F1-TRMJ*F2)
      RN2=RABS2-RDRDM*UBS2
      RRABD2 = -((TRMJ2-4.*BRM2)*F1/3.+BRM2*TRMJ*F2)
      RUBD2 = RABS2
      RVTBD2 = -((TRMJ2 + 2.*BRM2)*F1/3.-BRM2*TRMJ*F2)
      RRM2=RRABD2-RDRDM*RUBD2
C
      IF (J .NE. I) GO TO 1080
C
      T(J) = (RDRDX-RN1*T11)/RN2
      TC(J) = -(RS+RM1*TC11)/RRM2
 1080 RDRDX = RDRDX-RN2*T(J)
      RS=RS+RRM2*TC(J)
      IF(J.NE.1) GO TO 1090
      UBSUM = UBS1*T11
      UBDSUM = Q001
      VBSUM =  Q002
      VBDSUM = Q003
      VTDSUM = Q004
 1090 CONTINUE
      UBSUM = UBSUM+UBS2*T(J)
      VBSUM = VBSUM+RABS2*T(J)/RM
      UBDSUM = UBDSUM + RUBD2*TC(J)/RM
      VBDSUM = VBDSUM+RRABD2*TC(J)/RM2
      VTDSUM = VTDSUM+RVTBD2*TC(J)/RM2
      GO TO 1130
 1100 IF(I.EQ.J) GO TO 1120
      IF (J.NE.1) GO TO 1110
      PHIS = TRM1*T11
```

```
      PHID = 1.5*TRM1*TRM1*TC11
 1110 PHIS = PHIS+TRMJ2*T(J)
      PHID = PHID+TRMJ*TRMJ2*TC(J)
      GO TO 1130
 1120 T(J) = -PHIS/TRMJ2
      TC(J) =-PHID/(TRMJ*TRMJ2)
C
 1130 CONTINUE
C
      DO 1140 K=1,NTHETA
      UB(M,K) = UBSUM+UBDSUM*COSTB(K)
      VB(M,K) = VBSUM+VBDSUM*COSTB(K)
 1140 VTB(M,K) = VTDSUM*SINTB(K)
C
 1150 CONTINUE
C ******************
C
C
C     CALCULATION OF VELOCITY COMPONENTS IN THE FIELD
      IF (NWING)1160,1360,1160
 1160 CONTINUE
      IF (NACEL.NE.0) GO TO 1170
      NS=NBODY+1
      JM=1
      SY(1)=0.
      GO TO 1180
 1170 NS=1
      JM=2
      SY(1)=1.
      SY(2)=-1.
 1180 CONTINUE
      NPANEL=NBODY+NWING
      II=0
      DO 1350 I=NS,NPANEL
      II=II+1
      U(II)=0.
      VV(II)=0.
      WW(II)=0.
      AN1(II)=0.
      DO 1350 N=1,JM
      XCN=XC(I)-XN
      YCN=YC(I)-SY(N)*YN
      ZCN=ZC(I)-ZN
      IF (YCN.EQ.0.) GO TO 1190
      THETAA = ATAN2( YCN, ZCN )
      GO TO 1220
 1190 IF(ZCN)1210,1200,1200
 1200 THETAA=0.
      GO TO 1220
 1210 THETAA=PI
 1220 CONTINUE
      COSTHA=COS(THETAA)
      SINTHA=SIN(THETAA)
      R2=YCN*YCN+ZCN*ZCN
      IF(R2.LE.EPS) R2 = 0.
      R1=SQRT(R2)
      BR1=BETA*R1
      BR2=BT2*R2
      TRM=XCN-XB(NBODYS)+BETA*R(NBODYS)
      IF(R1.EQ.0.) GO TO 1240
      IF(TRM-BR1)1240,1240,1230
 1230 G1=TRM*TRM-BR2
      TRMO=TRM/BR1
      G2=ALOG(TRMO+SQRT(TRMO*TRMO-1.))
      G3=SQRT(G1)
```

```
1240 CONTINUE
     US=0.
     VS=0.
     UD=0.
     VD=0.
     VT=0.
     VTF = 0.
     VF = 0.
     IF (TRM.LE.BR1) GO TO 1250
      Q001 = 2.*G1*G3
      Q002 = R1*G3
      Q003 = G2*BR2
      Q004 = BR2/G1
      Q005 = 2.*R2
      Q006 = TC11*SINTHA
      Q007 = TRM*G3-BR2*G2
      Q008 = 2.*G3
      Q009 = 3.*G1*G3
      Q010 = 3.*G3
      Q011 = BT2*COSTHA*TC11
      Q012 = BT2*SINTHA*TC11
      Q013 = 2.*TRM
      Q014 = TC11*COSTHA
      Q017 = 2.*BR2
      Q018 = BT2*COSTHA
      Q019 = BT2*SINTHA
1250 DO 1340 J=1,N1
     BR=BETA*R(J)
     TRM1=XCN-XB(J)+BR
     TRM2=TRM1*TRM1
     IF(BR1.EQ.0.) GO TO 1270
     TRM3=TRM1/BR1
     IF(TRM3.LT.1.) GO TO 1340
C
C    VELOCITY DUE TO QUADRATIC SINGULARITIES
     F1=SQRT(TRM3*TRM3-1.)
     F2=ALOG(TRM3+F1)
     F1=BR1*F1
     US=2.*(F1-TRM1*F2)*T(J)
     VS=(TRM1*F1-BR2*F2)/R1
     UD=VS*TC(J)
     VS=VS*T(J)
     VD=-((TRM2-4.*BR2)*F1/3.+TRM1*BR2*F2)*TC(J)/R2
     VT=-((TRM2+2.*BR2)*F1/3.-TRM1*BR2*F2)*TC(J)/R2
C
C    VELOCITY DUE TO TANGENT CONE
     IF (J.NE.1) GO TO 1260
     US=US-F2*T11
     VS=VS+F1*T11/R1
     UD=UD+F1*TC11/R1
     VD=VD-.5*(TRM1*F1+BR2*F2)*TC11/R2
     VT=VT-.5*(TRM1*F1-BR2*F2)*TC11/R2
1260 CONTINUE
C
C    SUMMATION OF VELOCITY COMPONENTS
     U(II) = U(II) +US +COSTHA*UD
     VF = VF +VS +COSTHA*VD
     VTF = VTF +VT*SINTHA
1270 IF(TRM-BR1)1330,1330,1280
1280 XI = XB(NBODYS)-(XB(J)-BETA*R(J))-BETA*R(NBODYS)
     TRMX=TRM+XI
     IF(R1)1330,1310,1290
1290 IF(J.NE.1) GO TO 1300
     U(II)=U(II)+T11*(G2+XI/G3)-Q014       *(G3+XI*
    1(1.-XI*BR2/Q001       ))/R1
```

```
      VF=VF-T11*(TRM*TRMX-BR2)/Q002    +Q014        *(Q003
     1+TRMX*G3-XI*(G3-TRM*(2.+XI*(1.-Q004  )/G3)))/Q005
      VTF=VTF+Q006       *(Q007          +XI*TRM*(2.+XI/G3))/Q005
 1300 CONTINUE
      U(II)=U(II)+2.*T(J)*(TRMX*G2-G3+XI*XI/Q008   )
     1-TC(J)*((TRMX+XI)*G3-Q003  +XI*XI*(1.-XI*BR2/Q009       ))/R1
     2*COSTHA
      VF=VF-T(J)*((TRMX+XI)*G3-Q003  +TRM*XI*XI/G3)/R1
     1+TC(J)*COSTHA*((TRMX*TRMX-4.*BR2)*G3/3.+TRMX*Q003  +XI*(TRM-XI)
     2*G3/3.+XI*XI*TRM*(1.+XI*(1.-Q004  )/Q010   ))/R2
      VTF=VTF+TC(J)*SINTHA*((TRM2+Q017  )*G3/3.0
     1-TRMX*Q003  +XI*(TRM-XI)*G3/3.+XI*XI*TRM*(1.+XI/Q010   ))/R2
      GO TO 1330
 1310 IF(TRM.EQ.0.) GO TO 1330
      G4=ALOG(TRMX/TRM)
      IF(J.NE.1) GO TO 1320
      U(II)=U(II)+T11*(XI/TRM-G4)
      G5=-(G4+XI*XI/(2.*TRM2))/2.
      VF=VF+Q011          *G5
      VTF=VTF-Q012          *G5
 1320 U(II)=U(II)+2.*T(J)*(XI-TRMX*G4+XI*XI/Q013     )
      G6=TRMX*G4-XI*(1.-XI*XI/(6.*TRM2))
      VF=VF-Q018       *TC(J)*G6
      VTF=VTF+Q019       *TC(J)*G6
 1330 CONTINUE
 1340 CONTINUE
C
      VV(II) = VF*SINTHA+VTF*COSTHA+VV(II)
      WW(II) = VF*COSTHA-VTF*SINTHA+WW(II)
      THETAS=THETAA+THETA(I)
      AN1(II) = VF*COS(THETAS)-VTF*SIN(THETAS)+AN1(II)
C
 1350 CONTINUE
 1360 CONTINUE
C
C     CALCULATION OF PRESSURES, FORCES, AND MOMENTS
      BODYL=XB(NBODYS)-XB(1)
      CPSTAG = 1.42857 * ((1.+.2*XMACH2)**3.5 - 1.)/XMACH2
      ARJ = ARA
      SINARJ = SIN(ARJ)
      COSARJ = COS(ARJ)
      DO 1420 J=1,NTHETA
      DO 1410 I=1,NBODYS
      VPM = VTB(I,J) * COSTB(J)  + VB(I,J) * SINTB(J)
      WPM = VB(I,J) *  COSTB(J)  - VTB(I,J) * SINTB(J)
      UWPM  = UB(I,J)*COSARJ + WPM*SINARJ
      UWIND = 1. + UWPM
      VWIND = VPM
      WWIND = WPM * COSARJ  - UB(I,J) * SINARJ
      IF (CPCALC - 1.)  1370,1380,1390
 1370 CPBB(I,J) = -2. * UWPM
      GO TO 1400
 1380 XM2 = XMACH2 - 1.
      CPBB(I,J) = -2.*UWPM + XM2*UWPM**2 - VWIND**2 - WWIND**2
      GO TO 1400
 1390 Q2 = UWIND ** 2   + VWIND **2  + WWIND ** 2
      CPBB(I,J) = 1.42857*((1.+.2*XMACH2*(1.-Q2))**3.5-1.)/XMACH2
 1400 CONTINUE
 1410 CONTINUE
      IF(R(NBODYS).EQ.0.) CPBB(NBODYS,J)=CPSTAG
 1420 CONTINUE
C
      BBCL=0.
      BBCD=0.
      BBCM=0.
```

```
      IF (NS1 .EQ. 0) NS1 = 1
      IF (NS2 .EQ. 0) NS2 = NBODYS
      NBT = NS2 - NS1 + 1
      MM = 0
      DO 1440 M = NS1, NS2
      MM = MM + 1
      XBB(MM)=XB(M)
      RM=R(M)
      DO 1430 N=1,NTHETA
      CPB=CPBB(M,N)
      CP(N)=CPB
      YA(N)=CPB*COSTB(N)
 1430 YB(N)=YA(N)*COSTB(N)
      CALL SIMP(GM,THETAB,CP,NTHETA,IRR)
      CALL SIMP(DM,THETAB,YB,NTHETA,IRR)
      CALL SIMP(AM,THETAB,YA,NTHETA,IRR)
      IF(IRR.NE.1) GO TO 1570
      A(MM) = AM*RM
      D(MM) = DM*RM
      G(MM) = GM*RM
 1440 CONTINUE
      MM = 0
      DO 1450  N = NS1, NS2
      MM = MM + 1
      C(MM) = DZDX(N)*A(MM) + FD(N)*G(MM)
      IF (N .EQ. 1)  C(MM) = .5*CPSTAG*DS1
      IF (N .EQ. NBODYS .AND. R(NBODYS) .EQ. 0.)  C(MM) = .5*CPSTAG*DS2
 1450 CONTINUE
      CALL SIMP (CX,XBB,C,NBT,IRR)
      CALL SIMP (CZ,XBB,A,NBT,IRR)
      CN=-CZ
      IF(IRR.NE.1) GO TO 1570
      MM = 0
      DO 1460 N = NS1,NS2
      MM = MM + 1
      C(MM) = ZDELTA(N)*C(MM)
      G(MM) = XB(N) + XN - XP + R(N)*FD(N)
      IF (N .EQ. 1) G(MM) = XB(N) + XN   - XP + DS1/(2.*PI)
      IF (N.EQ.NBODYS.AND.R(NBODYS).EQ.0.) G(MM)=XB(N)+XN-XP+DS2/(2.*PI)
 1460 A(MM) = A(MM)*G(MM) + C(MM) + R(N)*DZDX(N)*D(MM)
      CALL SIMP (CM,XBB,A,NBT,IRR)
      IF(IRR.NE.1) GO TO 1570
      BBCL=CN*COS(ARA)-CX*SIN(ARA)
      BBCD=CX*COS(ARA)+CN*SIN(ARA)
      BBCM = CM + CX*(ZN-ZP)
      DO 1470 J=1,NTHETA
 1470 THETAB(J)=THETAB(J)*57.2957795
      DO 1475 N = 1, NBODYS
 1475 XB(N) = XB(N) + X1
C
C
C     OPTIONAL PRINT OF VELOCITY COMPONENTS
      IF (VOUT)1480,1550,1480
 1480 CONTINUE
      WRITE (NTAPEO,1600)T11,TC11,(XB(I),ZDELTA(I),R(I),FD(I),SD(I),
     1T(I),TC(I),I=1,N1)
      I = NBODYS
      WRITE (NTAPEO,1680)         XB(I),ZDELTA(I),R(I),FD(I),SD(I)
      IF(NACEL)1490,1500,1490
 1490 WRITE(NTAPEO,1670) NACEL,NACEL
      GO TO 1510
 1500 WRITE (NTAPEO,1610)
 1510 WRITE (NTAPEO,1640)
      WRITE (NTAPEO,1620) (THETAB(J),J=1,NTHETA)
      WRITE (NTAPEO,1630)
```

```
      DO 1520 I=1,NBODYS                                                00003790
 1520 WRITE (NTAPEO,1590) XB(I),(UB(I,J),J=1,NTHETA)                    00003800
      WRITE (NTAPEO,1650)                                               00003810
      WRITE (NTAPEO,1620) (THETAB(J),J=1,NTHETA)                        00003820
      WRITE (NTAPEO,1630)                                               00003830
      DO 1530 I=1,NBODYS                                                00003840
 1530 WRITE (NTAPEO,1590) XB(I),(VB(I,J),J=1,NTHETA)                    00003850
      WRITE (NTAPEO,1660)                                               00003860
      WRITE (NTAPEO,1620) (THETAB(J),J=1,NTHETA)                        00003870
      WRITE (NTAPEO,1630)                                               00003880
      DO 1540 I=1,NBODYS                                                00003890
 1540 WRITE (NTAPEO,1590) XB(I),(VTB(I,J),J=1,NTHETA)                   00003900
 1550 CONTINUE                                                          00003910
      WRITE (NTAPEC) NBODYS                                             00003920
      WRITE (NTAPEC) (XB(J),R(J),J=1,NBODYS)                            00003930
      WRITE (NTAPEC) (T(J),TC(J),J=1,NBODYS)                            00003940
      WRITE (NTAPEC) T11,TC11                                           00003950
 1560 CONTINUE                                                          00003960
C                                                                       00003970
C ------------------                                                    00003980
      RETURN                                                            00003990
C ------------------                                                    00004000
C                                                                       00004010
C                                                                       00004020
 1570 WRITE(NTAPEO,1690) IRR                                            00004030
      GO TO 1560                                                        00004040
 1580 FORMAT (53H****BODY POINT IS OUTSIDE MACH CONE - PROGRAM HALT IN  00004050
     118H SUBROUTINE KARMOR)                                            00004060
 1590 FORMAT(1H ,F14.4,F15.5,9F10.5/(F29.5,9F10.5))                     00004070
 1600 FORMAT(1H1//41H BODY GEOMETRY AND SOURCE STRENGTHS, T11=F8.5,5X,  00004080
     15HTC11=F8.5/56X,5HFIRST,9X,6HSECOND,11X,1HT,14X,2HTC/9X,          00004090
     29HX-STATION,7X,6HCAMBER,9X,6HRADIUS,2X,2(5X,10HDERIVATIVE)//      00004100
     3(1X,7F15.4))                                                      00004110
 1610 FORMAT(1H1//66H VELOCITY COMPONENTS ON BODY DUE TO BODY LINE SOURC00004120
     1ES AND DOUBLETS)                                                  00004130
 1620 FORMAT(1H0,4X,11HTHETA(DEG.),F14.4,9F10.4/(F29.4,9F10.4))         00004140
 1630 FORMAT(1H ,9X,1HX)                                                00004150
 1640 FORMAT(1H0,5X,8HAXIAL(U))                                         00004160
 1650 FORMAT(1H0,5X,10HRADIAL(VR))                                      00004170
 1660 FORMAT(1H0,5X,14HTANGENTIAL(VT))                                  00004180
 1670 FORMAT( 1H1,38H VELOCITY COMPONENTS ON NACELLE PAIR (,I2,23H) DUE 00004190
     1TO NACELLE PAIR (,I2,27H) LINE SOURCES AND DOUBLETS)              00004200
 1680 FORMAT(1H , 5F15.4)                                               00004210
 1690 FORMAT(1H0,18HINTEGRATION ERROR=I2)                               00004220
      END                                                                      -
CDECK FCALC
      OVERLAY(WANG, 7, 7)
      PROGRAM FCALC
C                                                                       00000020
C ..................................................................... 00000030
C     CONTROL SUBROUTINE FOR CALCULATION OF FORCES, MOMENTS             00000040
C     AND PRESSURES                                                     00000050
C ..................................................................... 00000060
C                                                                       00000070
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1          NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON /AVAR/  A(210),ACB(21),ABX(100),AWX(110),AREA(210),ARN(2),A00000100
     1               RWT(20),ALPHA(210),ALPHAB(210),ALPHAS(210),ALPHAT(100000110
     2               10),ALPHAW(110),ALPHAX(110),AWS(110),ABT(100),AN1N(00000120
     3               210,2),ALPHAA,ALPHAD,ARA,ARADEG,ARB,ARW,ARAS,AT,AAA00000130
      COMMON /BVAR/  B(210)                                             00000140
     1               ,BBCL,BBCD,BBCM                                    00000150
      COMMON/CVAR/C(210), CHORD(210),CL(210),CPNN(51,11,2)
     1               ,CLS(210)                                          00000170
     2               ,CDR,CASE,CPCALC,CBAR,CONSNT,CLBAR,CLX,CLM,CDM     00000180
```

```
      COMMON/DVAR/ D(210),DZDXB(51), DADEG,DARAD
      COMMON /IVAR/  IPOLAR                                             00000210
      COMMON /KVAR/  KASE,KONFIG,KPOLAR                                 00000220
      COMMON /NVAR/  NFMT(9),NROW(2)                                    00000230
     1         ,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,NTHETS,
     2         NXLE, NRG, NPOLR, NCLX
      COMMON /PVAR/  POLAR                                              00000260
      COMMON/RVAR/ R(51), RN(51,2), RATIOX, RFAREA
      COMMON /SVAR/  SEMIS,SLC                                          00000290
      COMMON/TVAR/ TITLE(20), THETA(210), THETAB(11),THETAS(11), THETAN
     1         (11,2), TCL(11), TCD(11), TCM(11), THICK,TWIST
      COMMON /UVAR/  UBWT(100),UWBT(110),UWWT(110),UNCL(210,2)          00000330
      COMMON /VVAR/  VBWT(100),VWBT(110),VWWT(110),VNCL(210,2)          00000340
     1         ,VOUT                                                    00000350
      COMMON /WVAR/  WBWT(100),WWBT(110),WWWT(110),WNCL(210,2)          00000360
      COMMON/XVAR/ XBAR(210), XC(210), XNI(2), XNXN(2), XNTN(2), XB(51),
     1         XN(51,2), XNNCD(2), XNNCL(2), XNNCM(2), XYZ(3), XCL(11),
     2         XCD(11), XNACEL, XP, XCPBAR
      COMMON /YVAR/  YBAR(210),YC(210),YNI(2)                           0000041C
      COMMON/ZVAR/ ZBAR(210), ZC(210), ZNI(2), ZDELTA(51), ZDN(51,2)
     1         ,ZP,ZA                                                   00000430
      COMMON /PANVEL/ VS1(110),VS2(110),WSS(110)                        00000435
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1         ,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH, CPB
      COMMON/WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
C                                                                       00000440
      DIMENSION  VVX(110),WVX(110),VWX(110),WWX(110)                    00000441
      DIMENSION VSS(110),WAS(110)                                       00000470
      DIMENSION ALPHAL(110),ALPHAU(110)                                 00000450
      DIMENSION CHORDL(20), CLBB(110),CLWW(110),CPB(110),CPU(110),
     1         CPL(110), CSHAPE(110)
      DIMENSION U(110),UBB(100),UBN(100),UBW(100),UWB(110),UWN(110),UWW(00000480
     1         110)                                                     00000490
      DIMENSION V(110),VBB(100),VBN(100),VBW(100),VWB(110),VWN(110),VWW(00000500
     1         110)                                                     00000510
      DIMENSION W(110),WBB(100),WBN(100),WBW(100),WWB(110),WWN(110),WWW(00000520
     1         110),WSCLL(20),WSCLU(20),WSCDL(20),WSCDU(20),WSCL(20),WS00000530
     2         CD(20)                                                   00000540
C                                                                       00000550
      INTEGER TITLE                                                     00000560
C                                                                       00000570
C                                                                       00000580
C ----------------------------------------------------------------------00000590
C     KONFIG = 1      WING ALONE CONFIGURATION                          00000600
C     KONFIG = 2      BODY ALONE CONFIGURATION                          00000610
C     KONFIG = 3      WING-BODY CONFIGURATION                           00000620
C ----------------------------------------------------------------------00000630
C                                                                       00000640
      BETA=SQRT(ABS(XMACH*XMACH-1.))                                    00000650
      BETA2=BETA*BETA                                                   00000660
      BETAM=2.4/1.414214*XMACH**4/BETA**.75                             00000670
      ARAXX=ARA                                                         00000671
      ARA=0.0                                                           00000672
      RATIOD=.75                                                        00000680
      IF(KONFIG.EQ.1) NR=NROW(1)                                        00000681
      IF(KONFIG.EQ.3) NR=NROW(2)                                        00000682
C                                                                       00000690
C     ANY CONFIGURATION                                                 00000700
      IF (KONFIG-2)1050,1000,1010                                       00000710
C                                                                       00000720
C     BODY-ALONE CONFIGURATION                                          00000730
 1000 CONTINUE                                                          00000740
      CALL FSF(10,NTAPEC,IRR)                                           00000750
      CALL FOR EOF (NTAPEC)                                             00000760
      REWIND NTAPEC                                                     00000770
```

```
      GO TO 1530
 1010 CONTINUE
C
C     WING-BODY CONFIGURATION
      DO 1040 J=1,NBODY
      IF(NACEL)1020,1020,1030
 1020 UBN(J)=0.
      VBN(J)=0.
      WBN(J)=0.
      GO TO 1040
 1030 UBN(J)=UNCL(J,1)+UNCL(J,2)
      VBN(J)=VNCL(J,1)+VNCL(J,2)
      WBN(J)=WNCL(J,1)+WNCL(J,2)
 1040 CONTINUE
C
C     WING ALONE OR WING-BODY CONFIGURATION
 1050 CONTINUE
      DO 1080 J=1,NWING
      UWB(J)=0.
      VWB(J)=0.
      WWB(J)=0.
      IF(NACEL)1060,1060,1070
 1060 UWN(J)=0.
      VWN(J)=0.
      WWN(J)=0.
      GO TO 1080
 1070 K=J+NBODY
      UWN(J)=UNCL(K,1)+UNCL(K,2)
      VWN(J)=VNCL(K,1)+VNCL(K,2)
      WWN(J)=WNCL(K,1)+WNCL(K,2)
 1080 CONTINUE
      IF(NACEL)1090,1090,1100
 1090 XNNCLT=0.
      XNNCDT=0.
      XNNCMT=0.
      GO TO 1120
 1100 DO 1110 J=1,NACEL
      XNNCL(J)=XNNCL(J)/RFAREA
      XNNCD(J)=XNNCD(J)/RFAREA
 1110 XNNCM(J)=XNNCM(J)/(CBAR*RFAREA)
      XNNCLT=XNNCL(1)+XNNCL(2)
      XNNCDT=XNNCD(1)+XNNCD(2)
      XNNCMT=XNNCM(1)+XNNCM(2)
 1120 CONTINUE
      IF (IPOLAR .NE. 0) GO TO 1180
      GO TO (1130,1180,1240),KASE
C
C     CALCULATE WING ALPHA, GIVEN WING CL     KASE = 1
 1130 CONTINUE
      IF (KONFIG-2)1140,1160,1160
C
C     WING ALONE CONFIGURATION
 1140 CONTINUE
      CALL CAMBW (NWING,NTAPEE,A,CL,ALPHAW)
      REWIND NTAPEE
      DO 1150 J=1,NWING
 1150 ALPHAW(J)=ALPHAW(J)+ALPHAX(J)
      GO TO 1320
C
C     WING-BODY CONFIGURATION
 1160 CONTINUE
      CALL FSF (1,NTAPEF,IRR)
      CALL DCPI (NBODY,NWING,NTAPEF,A,ABX,CL(NS),CL)
      REWIND NTAPEF
      CALL CAMBWB (NBODY,NWING,NTAPEE,NTAPEF,A,CL(NS),ABX,AWX)
```

```
      REWIND NTAPEE
      REWIND NTAPEF
      DO 1170 J=1,NWING
 1170 ALPHAW(J)=AWX(J)+ALPHAX(J)
      GO TO 1320
C
C     CALCULATE WING CL, GIVEN WING ALPHA     KASE = 2
 1180 CONTINUE
      IF (KONFIG-2)1190,1210,1210
C
C     WING ALONE CONFIGURATION
 1190 CONTINUE
      DO 1200 J=1,NWING
 1200 AWX(J)=ALPHAW(J)-ALPHAX(J)
      CALL FSF (1,NTAPEE,IRR)
      CALL DCPD(NWING,NTAPEE,A,AWX,CL)
      REWIND NTAPEE
      GO TO 1320
C
C     WING-BODY CONFIGURATION
 1210 CONTINUE
      DO 1220 J=1,NWING
 1220 AWX(J)=ALPHAW(J)-ALPHAX(J)
      DO 1230 J=1,NBODY
      READ (NTAPEF) (A(I),I=1,NWING)
      DO 1230 I=1,NWING
 1230 AWX(I)=AWX(I)-A(I)*ABX(J)
      CALL FSF (1,NTAPEE,IRR)
      CALL DCPD (NWING,NTAPEE,A,AWX,CL(NS))
      REWIND NTAPEE
      CALL FSF (1,NTAPEF,IRR)
      CALL DCPI (NBODY,NWING,NTAPEF,A,ABX,CL(NS),CL)
      REWIND NTAPEF
      GO TO 1320
C
C     OPTIMIZE CAMBER FOR GIVEN CL, OR FOR GIVEN CL AND CM     KASE = 3
 1240 CONTINUE
      IF (CONSNT)1260,1250,1260
 1250 CALL FSF(2,NTAPEA,IRR)
      GO TO 1270
 1260 CALL FSF(3,NTAPEA,IRR)
 1270 CONTINUE
      IF (KONFIG-2)1280,1300,1300
C
C     WING ALONE CONFIGURATION
 1280 CONTINUE
      CALL OPTMW (NWING,NTAPEA,A,B,CONSNT,CLBAR,XCPBAR,RFAREA,AREA,CL)
      REWIND NTAPEA
      CALL CAMBW (NWING,NTAPEE,A,CL,ALPHAW)
      REWIND NTAPEE
      DO 1290 J=1,NWING
 1290 ALPHAW(J)=ALPHAW(J)+ALPHAX(J)
      GO TO 1320
C
C     WING-BODY CONFIGURATION
 1300 CONTINUE
      CALL OPTMWB (NWING,NBODY,NTAPEA,NTAPEF,THICK,A,B,ALPHA,ABX,ALPHAX,
     1 AREA(NS),CONSNT,CLBAR,XCPBAR,RFAREA,CL(NS))
      REWIND NTAPEA
      CALL FSF (1,NTAPEF,IRR)
      CALL DCPI (NBODY,NWING,NTAPEF,A,ABX,CL(NS),CL)
      REWIND NTAPEF
      CALL CAMBWB (NBODY,NWING,NTAPEE,NTAPEF,A,CL(NS),ABX,AWX)
      REWIND NTAPEE
      REWIND NTAPEF
```

```
      DO 1310 J=1,NWING                                                 00002080
 1310 ALPHAW(J)=AWX(J)+ALPHAX(J)                                        00002090
C                                                                       00002100
C     WING ALONE OR WING-BODY CONFIGURATION                             00002110
 1320 CONTINUE                                                          00002120
      IF (KONFIG-2)1330,1340,1340                                       00002130
C                                                                       00002140
C     WING ALONE CONFIGURATION                                          00002150
 1330 CONTINUE                                                          00002160
      CALL FSF (1,NTAPEB,IRR)                                           00002170
      CALL CVEL (NWING,NWING,NTAPEB,A,B,C,CL,UWW,VWW,WWW)               00002180
      REWIND NTAPEB                                                     00002190
      GO TO 1350                                                        00002200
C                                                                       00002210
C     WING-BODY CONFIGURATION                                           00002220
 1340 CONTINUE                                                          00002230
      CALL FSF (1,NTAPED,IRR)                                           00002240
      CALL CVEL (NWING,NBODY,NTAPED,A,B,C,CL,UWB,VWB,WWB)               00002250
      CALL FSF (2,NTAPED,IRR)                                           00002260
      CALL CVEL (NWING,NWING,NTAPED,A,B,C,CL(NS),UWW,VWW,WWW)           00002270
      REWIND NTAPED                                                     00002280
      CALL RITE(NFMT(6),NTAPEO,NWING,NROWW,NCOLW,THETA,UWB,VWB,WWB)     0C002290
      CALL RITE(NFMT(7),NTAPEO,NWING,NROWW,NCOLW,THETA,UWBT,VWBT,WWBT)  00002300
C                                                                       00002310
C     WING ALONE OR WING-BODY CONFIGURATION                             00002320
 1350 CONTINUE                                                          00002330
      CALL FSF(10,NTAPEC,IRR)                                           00002340
      WRITE (NTAPEC) NPANEL                                             00002350
      WRITE (NTAPEC) (CL(J),J=1,NPANEL)                                 00002360
      CALL FOR EOF (NTAPEC)                                             00002370
      REWIND NTAPEC                                                     00002380
      DO 1345 J=1,NWING                                                 00002381
      JJ=J+NBODY                                                        00002382
      NJ=(J-1)/NR+1                                                     00002383
      VSS(J)=VS1(J)*CL(JJ)                                              00002384
      DO 1345 I=1,NWING                                                 00002385
      II=I+NBODY                                                        00002386
      NI=(I-1)/NR+1                                                     00002387
      IF(I.LT.J.AND.NI.EQ.NJ) VSS(J)=VSS(J)+VS2(I)*CL(II)               00002388
 1345 CONTINUE                                                          00002389
      DO 1360 J=1,NWING                                                 00002390
      JJ=J+NBODY                                                        00002400
      TJ=THETA(JJ)                                                      00002401
      COSTJ=COS(TJ)                                                     00002402
      SINTJ=SIN(TJ)                                                     00002403
      VVX(J)=VSS(J)*COSTJ                                               00002404
      WVX(J)=VSS(J)*SINTJ                                               00002405
      IF(THICK.EQ.0.) GO TO 1360                                        00002406
      WWX(J)=WSS(J)*COSTJ                                               00002407
      VWX(J)=WSS(J)*SINTJ                                               00002408
 1360 UWW(J)=UWW(J)-.25*CL(JJ)                                          00002410
      DO 1361 J=1,NWING                                                 00002412
      VWW(J)=VWW(J)+VVX(J)                                              00002413
 1361 WWW(J)=WWW(J)+WVX(J)                                              00002414
      CALL RITE(NFMT(4),NTAPEO,NWING,NROWW,NCOLW,THETA,UWW,VWW,WWW)     00002420
      DO 1365 J=1,NWING                                                 00002421
      JJ=J+NBODY                                                        00002422
      VWW(J)=VWW(J)-2.*VVX(J)                                           00002423
      WWW(J)=WWW(J)-2.*WVX(J)                                           00002424
 1365 UWW(J)=UWW(J)+.5*CL(JJ)                                           00002425
      CALL RITE(NFMT(4),NTAPEO,NWING,NROWW,NCOLW,THETA,UWW,VWW,WWW)     00002426
      DO 1366 J=1,NWING                                                 00002427
      JJ=J+NBODY                                                        00002428
      VWW(J)=VWW(J)+VVX(J)                                              00002429
      WWW(J)=WWW(J)+WVX(J)                                              00002430
```

```
 1366 UWW(J)=UWW(J)-.5*CL(JJ)                                          00002431
C     COMPUTE COEFFICIENT OF PRESSURE ON WING WITH OR WITHOUT WING      00002440
C     THICKNESS                                                         00002450
      CALL RITE(NFMT(5),NTAPEO,NWING,NROWW,NCOLW,THETA,UWWT,VWWT,WWWT)  00002451
      IF (THICK)1370,1410,1370                                          00002460
C                                                                       00002470
C     WING ALONE OR WING-BODY CONFIGURATION                             00002480
C     WITH WING THICKNESS                                               00002490
 1370 CONTINUE                                                          00002500
      DO 1375 J=1,NWING                                                 00002501
      VWWT(J)=VWWT(J)-VWX(J)                                            00002502
 1375 WWWT(J)=WWWT(J)+WWX(J)                                            00002503
      CALL RITE(NFMT(5),NTAPEO,NWING,NROWW,NCOLW,THETA,UWWT,VWWT,WWWT)  00002510
      DO 1376 J=1,NWING                                                 00002511
      VWWT(J)=VWWT(J)+2.*VWX(J)                                         00002512
 1376 WWWT(J)=WWWT(J)-2.*WWX(J)                                         00002513
      CALL RITE(NFMT(5),NTAPEO,NWING,NROWW,NCOLW,THETA,UWWT,VWWT,WWWT)  00002514
      DO 1377 J=1,NWING                                                 00002515
      VWWT(J)=VWWT(J)-VWX(J)                                            00002516
 1377 WWWT(J)=WWWT(J)+WWX(J)                                            00002517
      DO 1380 J=1,NWING                                                 00002520
      U(J)=UWB(J)+UWW(J)+UWBT(J)+UWWT(J)+UWN(J)                         00002530
      V(J)=VWB(J)+VWW(J)+VWBT(J)+VWWT(J)+VWN(J)-VWX(J)+VVX(J)           00002540
 1380 W(J)=WWB(J)+WWW(J)+WWBT(J)+WWWT(J)+WWN(J)+WWX(J)+WVX(J)           00002550
      CALL   CP(0,NWING,XMACH,CPCALC,U,V,W,CPU,ARA)                     00002560
      DO 1390 J=1,NWING                                                 00002570
      JJ=J+NBODY                                                        00002580
      U(J)=U(J)+.5*CL(JJ)                                               00002590
      V(J)=VWB(J)+VWW(J)+VWBT(J)+VWWT(J)+VWN(J)+VWX(J)-VVX(J)           00002600
 1390 W(J)=WWB(J)+WWW(J)+WWBT(J)+WWWT(J)+WWN(J)-WWX(J)-WVX(J)           00002610
      CALL   CP(0,NWING,XMACH,CPCALC,U,V,W,CPL,ARA)                     00002620
      DO 1400 J=1,NWING                                                 00002630
      ALPHAU(J)=ALPHAW(J)+WSS(J)                                        00003170
 1400 ALPHAL(J)=ALPHAW(J)-WSS(J)                                        00003180
      GO TO 1450                                                        00002660
C                                                                       00002670
C     WING ALONE OR WING-BODY CONFIGURATION                             00002680
C     WITHOUT WING THICKNESS                                            00002690
 1410 CONTINUE                                                          00002700
      DO 1420 J=1,NWING                                                 00002710
      U(J)=UWB(J)+UWW(J)+UWBT(J)+UWN(J)                                 00002720
      V(J)=VWB(J)+VWW(J)+VWBT(J)+VWN(J)+VVX(J)                          00002730
 1420 W(J)=WWB(J)+WWW(J)+WWBT(J)+WWN(J)+WVX(J)                          00002740
      CALL   CP(0,NWING,XMACH,CPCALC,U,V,W,CPU,ARA)                     00002750
      DO 1430 J=1,NWING                                                 00002760
      JJ=J+NBODY                                                        00002770
      V(J)=VWB(J)+VWW(J)+VWBT(J)+VWN(J)-VVX(J)                          00002780
      W(J)=WWB(J)+WWW(J)+WWBT(J)+WWN(J)-WVX(J)                          00002781
 1430 U(J)=U(J)+.5*CL(JJ)                                               00002790
      CALL   CP(0,NWING,XMACH,CPCALC,U,V,W,CPL,ARA)                     00002800
      DO 1440 J=1,NWING                                                 00002810
      ALPHAU(J)=ALPHAW(J)                                               00002820
 1440 ALPHAL(J)=ALPHAW(J)                                               00002830
C                                                                       00002840
C     WING ALONE OR WING-BODY CONFIGURATION                             00002850
C     CALCULATE CAMBER SHAPE                                            00002860
 1450 CONTINUE                                                          00002870
      DO 1451 J=1,NWING                                                 00003460
      JJ=1+(J-1)/NROWW                                                  00003470
      WAS(J)=ARWT(JJ)/CDR                                               00003480
 1451 WAS(J)=COS(WAS(J))*COS(ARAXX)                                     00003490
      CALL INTPOL(NWING,NROWW,RATIOX,.5,CHORD(NS),ALPHAW,ALPHA)         00002880
      CALL CAMBER(NCOLW,NROWW,CHORD(NS),ALPHA,CSHAPE,CHORDL)            00002890
C                                                                       00002900
C     COMPUTE COEFFICIENTS OF LIFT, DRAG, MOMENT ON WING                00002910
```

```
      CALL WLDM (NWING,NROWW,XP,ZP,RFAREA,AREA(NS),XBAR(NS),ZBAR(NS),   00003510
     X ALPHAU,THETA(NS),CPU,WSCLU,WSCDU,WCLU,WCDU,WCMU,YBAR(NS),WAS)    00003520
      CALL WLDM (NWING,NROWW,XP,ZP,RFAREA,AREA(NS),XBAR(NS),ZBAR(NS),   00003540
     X ALPHAL,THETA(NS),CPL,WSCLL,WSCDL,WCLL,WCDL,WCML,YBAR(NS),WAS)    00003550
      DO 1460 J=1,NCOLW                                                 00002980
      JN=(J-1)*NROWW+(NROWW+1)/2                                        00002990
      JN=NBODY+JN                                                       00003000
      DY=AREA(JN)/CHORD(JN)                                             00003010
      WSCD(J)=WSCDL(J)-WSCDU(J)                                         00003020
      WSCL(J)= WSCLL(J)-WSCLU(J)                                        00003030
 1460 CONTINUE                                                          00003040
      WCL=WCLU-WCLL                                                     00003050
      WCD=WCDU-WCDL                                                     00003060
 1465 CONTINUE                                                          00003066
      WCM=(WCMU-WCML)/CBAR                                              00003070
      IF (KONFIG .EQ. 1) GO TO 1540                                     00003080
C                                                                       00003090
C     WING-BODY CONFIGURATION                                           00003100
      CALL CVEL (NBODY,NBODY,NTAPED,A,B,C,CL,UBB,VBB,WBB)               00003110
      CALL FSF (2,NTAPED,IRR)                                           00003120
      CALL CVEL (NBODY,NWING,NTAPED,A,B,C,CL(NS),UBW,VBW,WBW)           00003130
      REWIND NTAPED                                                     00003140
      DO 1470 J=1,NBODY                                                 00003150
 1470 UBB(J)=UBB(J)-.25*CL(J)                                           00003160
      CALL RITE (NFMT(1),NTAPEO,NBODY,NROWB,NTHETA,THETAB,UBB,VBB,WBB)  00003170
      CALL RITE (NFMT(2),NTAPEO,NBODY,NROWB,NTHETA,THETAB,UBW,VBW,WBW)  00003180
C                                                                       00003190
C     COMPUTE COEFFICIENT OF PRESSURE ON BODY WITH OR WITHOUT THE       00003200
C     EFFECT OF WING THICKNESS                                          00003210
      IF (THICK)1480,1500,1480                                          00003220
 1480 CONTINUE                                                          00003230
      CALL RITE(NFMT(3),NTAPEO,NBODY,NROWB,NTHETA,THETAB,UBWT,VBWT,WBWT)00003240
      DO 1490 J=1,NBODY                                                 00003250
      U(J)=UBB(J)+UBW(J)+UBWT(J)+UBN(J)                                 00003260
      V(J)=VBB(J)+VBW(J)+VBWT(J)+VBN(J)                                 00003270
 1490 W(J)=WBB(J)+WBW(J)+WBWT(J)+WBN(J)                                 00003280
      GO TO 1520                                                        00003290
 1500 CONTINUE                                                          00003300
      DO 1510 J=1,NBODY                                                 00003310
      U(J)=UBB(J)+UBW(J)+UBN(J)                                         00003320
      V(J)=VBB(J)+VBW(J)+VBN(J)                                         00003330
 1510 W(J)=WBB(J)+WBW(J)+WBN(J)                                         00003340
 1520 CONTINUE                                                          00003350
      CALL  CP(1,NBODY,XMACH,CPCALC,U,V,W,CPB,ARA)                      00003360
C                                                                       00003370
C     COMPUTE COEFFICIENTS OF LIFT, DRAG, MOMENT ON BODY PANELS         00003380
      CALL INTPOL (NBODY,NROWB,RATIOX,RATIOD,CHORD,ALPHAB,ALPHA)        00003390
      CALL BLDM (NBODY,XP,ZP,RFAREA,AREA,XBAR,ZBAR,ALPHA,THETA,CPB      00003400
     1,BWCL,BWCD,BWCM)                                                  00003410
C                                                                       00003420
C     BODY ALONE OR WING-BODY CONFIGURATION                             00003430
 1530 CONTINUE                                                          00003440
      BBCL=BBCL/RFAREA                                                  00003450
      BBCD=BBCD/RFAREA                                                  00003460
      BBCM=BBCM/(RFAREA*CBAR)                                           00003470
C                                                                       00003480
      IF (KONFIG .EQ. 2) GO TO 1540                                     00003490
C                                                                       00003500
C     WING-BODY CONFIGURATION                                           00003510
      BWCM=BWCM/CBAR                                                    00003520
      BCL=BWCL+BBCL                                                     00003530
      BCD=BWCD+BBCD                                                     00003540
      BCM=BWCM+BBCM                                                     00003550
      WBCL=BCL+WCL+XNNCLT                                               00003560
      WBCD=BCD+WCD+XNNCDT                                               00003570
```

```
      WBCM=BCM+WCM+XNNCMT                                               00003580
C                                                                       00003590
C     ANY CONFIGURATION                                                 00003600
C     WRITE OUTPUT                                                      00003610
 1540 CONTINUE                                                          00003620
      ARA=ARAXX                                                         00003621
C                                                                       00003630
      IF (KONFIG-2)1630,1560,1550                                       00003640
C                                                                       00003650
C     WING-BODY CONFIGURATION                                           00003660
 1550 CONTINUE                                                          00003670
      WRITE (NTAPEO,1770)                                               00003680
      GO TO 1570                                                        00003690
C                                                                       00003700
C     BODY ALONE CONFIGURATION                                          00003710
 1560 CONTINUE                                                          00003720
      WRITE (NTAPEO,1820)                                               00003730
C                                                                       00003740
C     BODY ALONE OR WING-BODY CONFIGURATION                             00003750
 1570 CONTINUE                                                          00003760
      WRITE (NTAPEO,2010) BBCD,BBCL,BBCM                                00003770
      WRITE (NTAPEO,1830)                                               00003780
      WRITE(NTAPEO,1860) (THETB(I),I=1,NTHETB)                          00003790
      WRITE (NTAPEO,1870)                                               00003800
      IF(XMACH.LT.1.) GO TO 1590                                        00003810
      NBS = NBODYS
      DO 1580 I=1,NBODYS                                                00003820
      XMIDB(I) = XB(I)
 1580 WRITE (NTAPEO,1880)XB(I),(CPBB(I,J),J=1,NTHETB)                   00003830
      GO TO 1610                                                        00003840
 1590 NBS=NBODYS-1                                                      00003850
      DO 1600 I=1,NBS                                                   00003860
      XX=.5*(XB(I)+XB(I+1))                                             00003870
      XMIDB(I) = XX
 1600 WRITE (NTAPEO,1880) XX, (CPBB(I,J), J=1,NTHETB)                   00003880
 1610 IF(KONFIG.EQ.2) GO TO 1750                                        00003890
C                                                                       00003900
C     WING-BODY CONFIGURATION                                           00003910
      WRITE (NTAPEO,1790)                                               00003920
      WRITE (NTAPEO,2010) BWCD,BWCL,BWCM                                00003930
      WRITE (NTAPEO,1840)                                               00003940
      CALL OUTB (NTAPEO,NBODY,NTHETA,NROWB,THETAB,CPB)                  00003950
      WRITE (NTAPEO,1850)                                               00003960
      CALL OUTB (NTAPEO,NBODY,NTHETA,NROWB,THETAB,ALPHAB)               00003970
C                                                                       00003980
C     WING ALONE OR WING-BODY CONFIGURATION                             00003990
 1630 CONTINUE                                                          00004000
C                                                                       00004010
      IF (NACEL .EQ. 0) GO TO 1680                                      00004020
C                                                                       00004030
C     WING ALONE OR WING-BODY CONFIGURATION WITH NACELLE(S)             00004040
      DO 1670 J=1,NACEL                                                 00004050
      NTN=XNTN(J)                                                       00004060
      NXN=XNXN(J)                                                       00004070
      WRITE (NTAPEO,1890) J                                             00004080
      WRITE (NTAPEO,2010) XNNCD(J),XNNCL(J),XNNCM(J)                    00004090
      WRITE (NTAPEO,1900)                                               00004100
      WRITE (NTAPEO,1860) (THETAN(I,J),I=1,NTN)                         00004110
      WRITE (NTAPEO,1870)                                               00004120
      IF(XMACH.LT.1.) GO TO 1650                                        00004130
      DO 1640 I=1,NXN                                                   00004140
 1640 WRITE (NTAPEO,1880) XN(I,J),(CPNN(I,K,J),K=1,NTN)                 00004150
      GO TO 1670                                                        00004160
 1650 NXS=NXN-1                                                         00004170
      DO 1660 I=1,NXS                                                   00004180
```

```
      XX=.5*(XN(I,J)+XN(I+1,J))
 1660 WRITE(NTAPEO,1880)   XX, (CPNN(I,K,J),K=1,NTN)
 1670 CONTINUE
C
 1680 CONTINUE
      DO 1690 J=NS,NPANEL
 1690 CL(J)=-CL(J)
C
      IF (KONFIG-2)1700,1710,1710
C
C     WING ALONE CONFIGURATION
 1700 CONTINUE
      WRITE (NTAPEO,1810)
      GO TO 1720
C
C     WING-BODY CONFIGURATION
 1710 CONTINUE
      WRITE (NTAPEO,1780)
C
C     WING ALONE OR WING-BODY CONFIGURATION
 1720 CONTINUE
      WRITE (NTAPEO,2010) WCD,WCL,WCM
      WRITE (NTAPEO,1980)
      WRITE (NTAPEO,2000) (I,I=1,NCOLW)
      WRITE (NTAPEO,1760) (WSCD(I),I=1,NCOLW)
      WRITE (NTAPEO,1990)
      WRITE (NTAPEO,2000) (I,I=1,NCOLW)
      WRITE (NTAPEO,1760) (WSCL(I),I=1,NCOLW)
      WRITE (NTAPEO,1910)
      CALL OUTW (NTAPEO,NWING,NCOLW,NROWW,CPU)
      WRITE (NTAPEO,1920)
      CALL OUTW (NTAPEO,NWING,NCOLW,NROWW,CPL)
      WRITE (NTAPEO,1930)
      CALL OUTW (NTAPEO,NWING,NCOLW,NROWW,CL(NS))
      IF (THICK)1730,1740,1730
 1730 WRITE (NTAPEO,1950)
      CALL OUTW (NTAPEO,NWING,NCOLW,NROWW,ALPHAU)
      WRITE (NTAPEO,1960)
      CALL OUTW (NTAPEO,NWING,NCOLW,NROWW,ALPHAL)
 1740 WRITE (NTAPEO,1970)
      CALL OUTW (NTAPEO,NWING,NCOLW,NROWW,ALPHAW)
      WRITE (NTAPEO,1940)
      WRITE (NTAPEO,1760)(CHORDL(I),I=1,NCOLW)
C
      IF (KONFIG .EQ. 1) GO TO 1750
C
C     WING-BODY CONFIGURATION
      WRITE (NTAPEO,1800)
      WRITE (NTAPEO,2010) WBCD,WBCL,WBCM
C
C     ANY CONFIGURATION
 1750 CONTINUE
C
C
C              REDUCE POLAR BY 1 FOR EACH PASS
C
      POLAR = POLAR - 1
C ------------------
C ------------------
C
C
C     FORMAT STATEMENTS
 1760 FORMAT(1H ,F29.5,9F10.5)
 1770 FORMAT(1H1, 48H PRESSURES, FORCES, AND MOMENTS ON ISOLATED BODY)
 1780 FORMAT(1H1, 66H PRESSURES, FORCES, AND MOMENTS ON WING PANELS IN P
```

AD-A048 840 MCDONNELL DOUGLAS ASTRONAUTICS CO HUNTINGTON BEACH CALIF F/G 1/1
AERODYNAMIC COMPUTER CODE FOR COMPUTING PRESSURE LOADING ON COM--ETC(U)
JAN 78 K K WANG

UNCLASSIFIED MDC-G7215 NL

3 OF 3
ADA048840

END
DATE
FILMED
2-78
DDC

```
     1RESENCE OF BODY)                                                  00004810
 1790 FORMAT(1H1, 70H INCREMENTAL PRESSURES, FORCES, AND MOMENTS ON BODY00004820
     1 PANELS DUE TO WING)                                              00004830
 1800 FORMAT(1H1, 44H FORCES AND MOMENTS ON WING-BODY COMBINATION)      00004840
 1810 FORMAT(1H1, 38H PRESSURES, FORCES AND MOMENTS ON WING)            00004850
 1820 FORMAT(1H1, 38H PRESSURES, FORCES AND MOMENTS ON BODY)            00004860
 1830 FORMAT(1H0/32H  BODY PRESSURE COEFFICIENTS(CP))                   00004870
 1840 FORMAT(1H0/37H  BODY PANEL PRESSURE COEFFICIENT(CP))              00004880
 1850 FORMAT(1H0/25H  BODY PANEL SLOPE(DR/DX))                          00004890
 1860 FORMAT(1H0,4X,11HTHETA(DEG.),F14.4,9F10.4,/(F29.4,9F10.4))        00004900
 1870 FORMAT(1H ,9X,1HX)                                                00004910
 1880 FORMAT(1H ,F14.4,F15.5,9F10.5,/(F29.5,9F10.5))                    00004920
 1890 FORMAT(1H1, 57H PRESSURES, FORCES, AND MOMENTS ON ISOLATED NACELLE00004930
     1 PAIR ,I2)                                                        00004940
 1900 FORMAT(1H0/34H NACELLE PRESSURE COEFFICIENTS(CP))                 00004950
 1910 FORMAT(1H0/52H  UPPER SURFACE WING PANEL PRESSURE COEFFICIENTS(CP)00004960
     1)                                                                 00004970
 1920 FORMAT(1H0/52H  LOWER SURFACE WING PANEL PRESSURE COEFFICIENTS(CP)00004980
     1)                                                                 00004990
 1930 FORMAT(1H0/36H  WING PANEL PRESSURE DIFFERENCE(CL))               00005000
 1940 FORMAT(1H0/23H  WING CHORD LENGTHS(C))                            00005010
 1950 FORMAT(1H0/39H  UPPER SURFACE WING PANEL SLOPE(DZ/DX))            00005020
 1960 FORMAT(1H0/39H  LOWER SURFACE WING PANEL SLOPE(DZ/DX))            00005030
 1970 FORMAT(1H0/26H  WING CAMBER SLOPE(DZ/DX))                         00005040
 1980 FORMAT(1H0/25H  SECTION CD DISTRIBUTION)                          00005050
 1990 FORMAT(1H0/25H  SECTION CL DISTRIBUTION)                          00005060
 2000 FORMAT(1H0,2X,16HSPANWISE STATION,I8,9I10,/(18X,I8,9I10))         00005070
 2010 FORMAT(1H0/6H  CD =,F10.5,10X,4HCL =,F10.5,10X,4HCM =,F10.5)      00005080
      END                                                                      -
CDECK OUTW
      SUBROUTINE OUTW (NTAPEO,NWING,NCOLW,NROWW,Z)                      OUTW1000
C                                                                       OUTW1010
C ..................................................................... OUTW1020
C     WING OUPUT FORMAT                                                 OUTW1030
C ..................................................................... OUTW1040
C                                                                       OUTW1050
      DIMENSION Z(1)                                                    OUTW1060
C                                                                       OUTW1070
      WRITE (NTAPEO,1010) (I,I=1,NCOLW)                                 OUTW1080
      WRITE (NTAPEO,1020)                                               OUTW1090
      DO 1000 J=1,NROWW                                                 OUTW1100
 1000 WRITE (NTAPEO,1030) (J,(Z(I),I=J,NWING,NROWW))                    OUTW1110
C                                                                       OUTW1120
C ------------------                                                    OUTW1130
      RETURN                                                            OUTW1140
C ------------------                                                    OUTW1150
 1010 FORMAT(1H0,2X,16HSPANWISE STATION,I8,9I10/(18X,I8,9I10))          OUTW1160
 1020 FORMAT(1H ,2X,17HCHORDWISE STATION)                               OUTW1170
 1030 FORMAT(1H ,I10,F19.5,9F10.5/(F29.5,9F10.5))                       OUTW1180
      END                                                                      -
CDECK OUTB
      SUBROUTINE OUTB (NTAPEO,NBODY,NTHETA,NROWB,THETA,Z)               OUTB1000
C                                                                       OUTB1010
C ..................................................................... OUTB1020
C     BODY OUTPUT FORMAT                                                OUTB1030
C ..................................................................... OUTB1040
C                                                                       OUTB1050
      DIMENSION Z(1),THETA(1)                                           OUTB1060
C                                                                       OUTB1070
      WRITE (NTAPEO,1010) (THETA(I),I=1,NTHETA)                         OUTB1080
      WRITE (NTAPEO,1020)                                               OUTB1090
      DO 1000 J=1,NROWB                                                 OUTB1100
 1000 WRITE (NTAPEO,1030) (J,(Z(I),I=J,NBODY,NROWB))                    OUTB1110
C                                                                       OUTB1120
C ------------------                                                    OUTB1130
```

```
      RETURN                                                            OUTB1140
C ------------------                                                    OUTB1150
 1010 FORMAT(1H0,4X,11HTHETA(DEG.),F14.4,9F10.4/(F29.4,9F10.4))         OUTB1160
 1020 FORMAT(1H0,6X,7HROW NO.)                                          OUTB1170
 1030 FORMAT(1H ,I10,F19.5,9F10.5/(F29.5,9F10.5))                       OUTB1180
      END                                                                      -
CDECK BLDM
      SUBROUTINE BLDM (NM,XP,ZP,RFAREA,AREA,XBAR,ZBAR,ALPHAM,THETAM     BLDM1000
     1,CPM,CL,CD,CM)                                                    BLDM1010
C                                                                       BLDM1020
C ..................................................................... BLDM1030
C     COMPUTES COEFFICIENT OF LIFT, DRAG, AND MOMENT ON BODY            BLDM1040
C ..................................................................... BLDM1050
C                                                                       BLDM1060
      DIMENSION AREA(1),XBAR(1),ZBAR(1),ALPHAM(1),THETAM(1),CPM(1)      BLDM1070
C                                                                       BLDM1080
      DRAG=0.                                                           BLDM1090
      XLIFT=0.                                                          BLDM1100
      XM=0.0                                                            BLDM1110
C                                                                       BLDM1120
      DO 1000 J=1,NM                                                    BLDM1130
      F=CPM(J)*AREA(J)                                                  BLDM1140
      XL=-F*COS(THETAM(J))                                              BLDM1150
      XD=F*ALPHAM(J)                                                    BLDM1160
      XM=XM-XL*(XBAR(J)-XP)+XD*(ZBAR(J)-ZP)                             BLDM1170
      XLIFT=XLIFT+XL                                                    BLDM1180
      DRAG=DRAG+XD                                                      BLDM1190
 1000 CONTINUE                                                          BLDM1200
C                                                                       BLDM1210
      CL=XLIFT/RFAREA                                                   BLDM1220
      CD=DRAG/RFAREA                                                    BLDM1230
      CM=XM/RFAREA                                                      BLDM1240
C                                                                       BLDM1250
C ------------------                                                    BLDM1260
      RETURN                                                            BLDM1270
C ------------------                                                    BLDM1280
      END                                                                      -
CDECK WLDM
      SUBROUTINE WLDM (NM,NROW,XP,ZP,RFAREA,AREA,XBAR,ZBAR,ALPHAM,      00000010
     X THETAM,CPM,SCL,SCD,CL,CD,CM,YBAR,AWS)                            00000020
C                                                                       00000030
C ..................................................................... 00000040
C     COMPUTES COEFFICIENT OF LIFT, DRAG, AND MOMENT ON WING            00000050
C     SPANWISE DISTRIBUTION OF LIFT AND DRAG COEFFICIENTS               00000060
C ..................................................................... 00000070
C                                                                       00000080
      DIMENSION AREA(1),XBAR(1),ZBAR(1),ALPHAM(1),THETAM(1),CPM(1)      00000090
     1,SCL(1),SCD(1),YBAR(1)                                            00000100
      DIMENSION AWS(1)                                                  00000105
C                                                                       00000110
      PI=3.14159                                                        00000111
      NCOL=NM/NROW                                                      00000120
      WDRAG=0.                                                          00000130
      WLIFT=0.                                                          00000140
      WMOM=0.                                                           00000150
      J=0                                                               00000160
      DO 1010 K=1,NCOL                                                  00000170
      DRAG=0.                                                           00000180
      XLIFT=0.                                                          00000190
      SUM=0.0                                                           00000200
      CON=1.0                                                           00000201
C                                                                       00000210
      DO 1000 I=1,NROW                                                  00000220
      J=J+1                                                             00000230
      IF(I.EQ.1) TEST=ABS(YBAR(J))+ABS(ABS(THETAM(J))-.5*PI)            00000231
```

```
      IF(TEST.LT.0.001) CON=.5                                          00000232
      SUM=AREA(J)+SUM                                                   00000240
      F=CPM(J)*AREA(J)                                                  00000250
      XL=-F*COS(THETAM(J))*CON*AWS(J)                                   00000300
      XD=F*SIN(ALPHAM(J))*CON                                           00000310
      WMOM=WMOM-XL*(XBAR(J)-XP)+XD*(ZBAR(J)-ZP)                         00000280
      XLIFT=XLIFT+XL                                                    00000290
      DRAG=DRAG+XD                                                      00000300
 1000 CONTINUE                                                          00000310
C                                                                       00000320
      SCD(K)=-DRAG/SUM                                                  00000330
      SCL(K)=-XLIFT/SUM                                                 00000340
      WDRAG=WDRAG+DRAG                                                  00000350
      WLIFT=WLIFT+XLIFT                                                 00000360
 1010 CONTINUE                                                          00000370
C                                                                       00000380
      CL=WLIFT/RFAREA                                                   00000390
      CD=WDRAG/RFAREA                                                   00000400
      CM=WMOM/RFAREA                                                    00000410
C                                                                       00000420
C ------------------                                                    00000430
      RETURN                                                            00000440
C ------------------                                                    00000450
      END                                                                      -
CDECK CAMBER
      SUBROUTINE CAMBER(NCOLW,NROWW,CHORD,ALPHAW,CSHAPE,CHORDL)         CAMB1000
C                                                                       CAMB1010
C ..................................................................... CAMB1020
C     COMPUTES SURFACE SHAPE, GIVEN SURFACE SLOPES                      CAMB1030
C ..................................................................... CAMB1040
C                                                                       CAMB1050
      DIMENSION CHORD(1),ALPHAW(1),CSHAPE(1),CHORDL(1)                  CAMB1060
C                                                                       CAMB1070
      DO 1020 J=1,NCOLW                                                 CAMB1080
      CHORDL(J)=0.                                                      CAMB1090
      DO 1000 I=1,NROWW                                                 CAMB1100
      K=(J-1)*NROWW+I                                                   CAMB1110
 1000 CHORDL(J)=CHORDL(J)+CHORD(K)                                      CAMB1120
      DO 1020 I=1,NROWW                                                 CAMB1130
      SUM=0.                                                            CAMB1140
      IW=I                                                              CAMB1150
      DO 1010 IJ=1,IW                                                   CAMB1160
      K=(J-1)*NROWW+IJ                                                  CAMB1170
 1010 SUM=SUM+ALPHAW(K)*CHORD(K)/CHORDL(J)                              CAMB1180
      CSHAPE(K)=SUM                                                     CAMB1190
 1020 CONTINUE                                                          CAMB1200
C                                                                       CAMB1210
C ------------------                                                    CAMB1220
      RETURN                                                            CAMB1230
C ------------------                                                    CAMB1240
      END                                                                      -
      SUBROUTINE INTPOL (NM,NROW,RATIOX,RATIOD,CHORD,ALPHAM,ALPHAD)     INTP1000
C                                                                       INTP1010
C ..................................................................... INTP1020
C     INTERPOLATES FOR NORMAL VELOCITY COMPONENT AT A SPECIFIED PERCENT INTP1030
C     OF PANEL CHORD                                                    INTP1040
C ..................................................................... INTP1050
C                                                                       INTP1060
      DIMENSION CHORD(1),ALPHAM(1),ALPHAD(1)                            INTP1070
C                                                                       INTP1080
      NCOL=NM/NROW                                                      INTP1090
      R=RATIOD-RATIOX                                                   INTP1100
      RR=-R                                                             INTP1110
      J=0                                                               INTP1120
C                                                                       INTP1130
```

```
      DO 1000 K=1,NCOL
      J=J+1
      ALPHAD(J)=ALPHAM(J)+R*(ALPHAM(J+1)-ALPHAM(J))/(1.+RATIOX
     1*(CHORD(J+1)/CHORD(J)-1.))
      DO 1000 I=2,NROW
      J=J+1
      CR=CHORD(J)/CHORD(J-1)
      ALPHAD(J)=ALPHAM(J)+CR*R*(ALPHAM(J)-ALPHAM(J-1))/(1.+
     1RATIOX*(CR-1.))
 1000 CONTINUE
C
C ------------------
      RETURN
C ------------------
      END
CDECK CP
      SUBROUTINE CP (K,NP,XMACH,CPCALC,U,V,W,CPP,ARA)
C
C ................................................................
C     COMPUTES  LINEAR, NONLINEAR, OR EXACT CP
C ................................................................
C
      DIMENSION U(1),V(1),W(1),CPP(1)
C
      XM2=XMACH*XMACH
      BT2    =  XM2 - 1.
      COSARA = COS(ARA)
      SINARA = SIN(ARA)
      IF(XM2.EQ.0.) GO TO 1000
      Q000   =   1.42857/XM2
      Q001   =   0.2 * XM2
 1000 DO 1050 J=1,NP
      UWPM =  U(J) * COSARA  + W(J) * SINARA
      UWIND = 1. + UWPM
      VWIND  = V(J)
      WWIND  = W(J) * COSARA  -  U(J) * SINARA
      IF (CPCALC - 1.) 1010,1020,1030
 1010 CPP(J) = - 2. * UWPM
      GO TO 1050
 1020 CPP(J) = -2.*UWPM  + BT2 * UWPM**2 - VWIND**2 - WWIND**2
      GO TO 1050
 1030 Q2  =  UWIND**2 + VWIND**2 + WWIND**2
      IF ( XMACH .EQ. 0.)  GO TO 1040
      CPP(J) = Q000*((1.+Q001*(1.-Q2))**3.5-1.)
      GO TO 1050
 1040 CPP(J) = 1. - Q2
 1050 CONTINUE
C
C ------------------
      RETURN
C ------------------
      END
CDECK RITE
      SUBROUTINE RITE (NFMT,NTAPEO,NP,NROW,NCOL,THETA,U,V,W)
C     PRINT OUT VELOCITY COMPONENTS
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1           SYM, KACE, NPOLAR, IRW
      COMMON/TVAR/ T1(20),THET(210),T2(77),THICK,TWIST
      DIMENSION U(1),V(1),W(1),THETA(1),AN(200)
      IF(NFMT-9) 2,2,3
    2 IF(NFMT) 3,999,5
    3 WRITE (NTAPEO,6070) NFMT
      GO TO 999
    5 CONTINUE
      NTEST=NFMT
```

```
      NWRIT=NP
      DO 8 N=1,NP
      IF(NTEST.LE.3) GO TO 6
      NN=N+NBODY
      AN(N)=W(N)*COS(THETA(NN))-V(N)*SIN(THETA(NN))
      GO TO 8
    6 THETB=THET(N)
      AN(N)=W(N)*COS(THETB)-V(N)*SIN(THETB)
    8 CONTINUE
      IGO=NFMT
      GO TO(10,20,30,40,50,60,70,300,200),IGO
   10 WRITE (NTAPEO,6030)
      GO TO 200
   20 WRITE (NTAPEO,6040)
      GO TO 200
   30 WRITE (NTAPEO,6045)
      GO TO 200
   40 WRITE (NTAPEO,6050)
      GO TO 300
   50 WRITE (NTAPEO,6055)
      GO TO 300
   60 WRITE (NTAPEO,6060)
      GO TO 300
   70 WRITE (NTAPEO,6065)
      GO TO 300
C     PRINT OUT VELOCITY COMPONENTS AT BODY PANELS
  200 CONTINUE
      WRITE (NTAPEO,6010)
      CALL OUTB (NTAPEO,NP,NCOL,NROW,THETA,U)
      WRITE (NTAPEO,6015)
      CALL OUTB (NTAPEO,NP,NCOL,NROW,THETA,V)
      WRITE (NTAPEO,6020)
      CALL OUTB (NTAPEO,NP,NCOL,NROW,THETA,W)
      WRITE(NTAPEO,6025)
      CALL OUTB (NTAPEO,NWRIT,NCOL,NROW,THETA,AN)
      GO TO 999
C     PRINT OUT VELOCITY COMPONENTS AT WING PANELS
  300 CONTINUE
      WRITE (NTAPEO,6010)
      CALL OUTW (NTAPEO,NP,NCOL,NROW,U)
      WRITE (NTAPEO,6015)
      CALL OUTW (NTAPEO,NP,NCOL,NROW,V)
      WRITE (NTAPEO,6020)
      CALL OUTW (NTAPEO,NP,NCOL,NROW,W)
      WRITE(NTAPEO,6025)
      CALL OUTW (NTAPEO,NWRIT,NCOL,NROW,AN)
 6000 FORMAT(1H ,10F11.5)
 6010 FORMAT(1H0,5X,8HAXIAL(U))
 6015 FORMAT(1H0,5X,13HTRANSVERSE(V))
 6020 FORMAT(1H0,5X,11HVERTICAL(W))
 6025 FORMAT(1H0,5X,9HNORMAL(N))
 6030 FORMAT(1H1, 76H VELOCITY COMPONENTS ON BODY PANELS DUE TO BODY PAN
     1EL PRESSURE SINGULARITIES)
 6040 FORMAT(1H1, 76H VELOCITY COMPONENTS ON BODY PANELS DUE TO WING PAN
     1EL PRESSURE SINGULARITIES)
 6045 FORMAT(1H1, 55H VELOCITY COMPONENTS ON BODY PANELS DUE TO WING SOU
     1RCES)
 6050 FORMAT(1H1, 76H VELOCITY COMPONENTS ON WING PANELS DUE TO WING PAN
     1EL PRESSURE SINGULARITIES)
 6055 FORMAT(1H1, 55H VELOCITY COMPONENTS ON WING PANELS DUE TO WING SOU
     1RCES)
 6060 FORMAT(1H1, 76H VELOCITY COMPONENTS ON WING PANELS DUE TO BODY PAN
     1EL PRESSURE SINGULARITIES)
 6065 FORMAT(1H1, 73H VELOCITY COMPONENTS ON WING PANELS DUE TO BODY LIN
     1E SOURCES AND DOUBLETS)
```

```
 6070 FORMAT(1H0,5HNFMT=,I10,18HIN SUBROUTINE RITE)
  999 CONTINUE                                                          066
      RETURN                                                            067
      END                                                                 -
CDECK CVEL
      SUBROUTINE CVEL(M,N,NTAPEX,A,B,C,CL,U,V,W)                       CVEL1000
C                                                                      CVEL1010
C .................................................................... CVEL1020
C     COMPUTES VELOCITY COMPONENTS FOR A GIVEN PANEL PRESSURE DIFFERENCECVEL1030
C .................................................................... CVEL1040
C                                                                      CVEL1050
      DIMENSION A(1),B(1),C(1),U(1),V(1),W(1),CL(1)                    CVEL1060
      DO 1000 J=1,M                                                    CVEL1070
      U(J)=0.0                                                         CVEL1080
      V(J)=0.0                                                         CVEL1090
      W(J)=0.0                                                         CVEL1100
 1000 CONTINUE                                                         CVEL1110
C                                                                      CVEL1120
      DO 1010 J=1,N                                                    CVEL1130
      READ (NTAPEX) (A(I),B(I),C(I),I=1,M)                             CVEL1140
      DO 1010 I=1,M                                                    CVEL1150
      U(I)=U(I)+A(I)*CL(J)                                             CVEL1160
      V(I)=V(I)+B(I)*CL(J)                                             CVEL1170
      W(I)=W(I)+C(I)*CL(J)                                             CVEL1180
 1010 CONTINUE                                                         CVEL1190
C                                                                      CVEL1200
C ------------------                                                   CVEL1210
      RETURN                                                           CVEL1220
C ------------------                                                   CVEL1230
      END                                                                 -
      SUBROUTINE OPTMWB (NW,NB,NTAPEX,NTAPEY,THICK,A,B,ALPHA,ABX,ALPHAX,OPTM1000
     1 AREA,CONSNT,CLBAR,XCPBAR,RFAREA,CL)                             OPTM1010
C                                                                      OPTM1020
C .................................................................... OPTM1030
C     OPTIMIZE WING-BODY CASE                                          OPTM1040
C .................................................................... OPTM1050
C                                                                      OPTM1060
      DIMENSION A(1),B(1),ALPHA(1),ALPHAX(1),AREA(1),CL(1)             OPTM1070
      DIMENSION ABX(1)                                                 OPTM1080
C                                                                      OPTM1090
      IF (THICK)1020,1000,1020                                         OPTM1100
 1000 CONTINUE                                                         OPTM1110
      DO 1010 I=1,NB                                                   OPTM1120
 1010 B(I)=0.                                                          OPTM1130
      GO TO 1040                                                       OPTM1140
 1020 CONTINUE                                                         OPTM1150
      DO 1030 I=1,NB                                                   OPTM1160
 1030 B(I)=-ABX(I)                                                     OPTM1170
 1040 CONTINUE                                                         OPTM1180
      DO 1050 J=1,NW                                                   OPTM1190
 1050 ALPHA(J)=ALPHAX(J)                                               OPTM1200
      DO 1060 J=1,NB                                                   OPTM1210
      READ (NTAPEY) (A(I),I=1,NW)                                      OPTM1220
      DO 1060 I=1,NW                                                   OPTM1230
 1060 ALPHA(I)=ALPHA(I)-A(I)*B(J)                                      OPTM1240
      DO 1070 J=1,NW                                                   OPTM1250
 1070 B(J)=AREA(J)*ALPHA(J)                                            OPTM1260
      NN=NW+1                                                          OPTM1270
      B(NN)=CLBAR*RFAREA                                               OPTM1280
C                                                                      OPTM1290
C     TEST IF PITCHING MOMENT CONSTRAINT                               OPTM1300
      IF (CONSNT)1080,1090,1080                                        OPTM1310
 1080 CONTINUE                                                         OPTM1320
      NN=NW+2                                                          OPTM1330
      B(NN)=XCPBAR*CLBAR*RFAREA                                        OPTM1340
```

```
 1090 CONTINUE                                                          OPTM1350
      DO 1100 J=1,NW                                                    OPTM1360
 1100 CL(J)=0.                                                          OPTM1370
      DO 1110 J=1,NN                                                    OPTM1380
      READ (NTAPEX) (A(I),I=1,NN)                                       OPTM1390
      DO 1110 I=1,NW                                                    OPTM1400
 1110 CL(I)=CL(I)+A(I)*B(J)                                             OPTM1410
C                                                                       OPTM1420
C ------------------                                                    OPTM1430
      RETURN                                                            OPTM1440
C ------------------                                                    OPTM1450
      END                                                                      -
CDECK OPTMW
      SUBROUTINE OPTMW (NW,NTAPEX,A,B,CONSNT,CLBAR,XCPBAR,RFAREA,AREA,CLOPTM1000
     1)                                                                 OPTM1010
C                                                                       OPTM1020
C ..................................................................... OPTM1030
C     OPTIMIZE WING ONLY CASE                                           OPTM1040
C ..................................................................... OPTM1050
C                                                                       OPTM1060
      DIMENSION A(1),B(1),AREA(1),CL(1)                                 OPTM1070
C                                                                       OPTM1080
      NN=NW+1                                                           OPTM1090
      B(NN)=CLBAR*RFAREA                                                OPTM1100
C                                                                       OPTM1110
C     TEST IF PITCHING MOMENT CONSTRAINT                                OPTM1120
      IF (CONSNT)1000,1010,1000                                         OPTM1130
 1000 CONTINUE                                                          OPTM1140
      NN=NW+2                                                           OPTM1150
      B(NN)=XCPBAR*CLBAR*RFAREA                                         OPTM1160
 1010 CONTINUE                                                          OPTM1170
      DO 1020 J=1,NW                                                    OPTM1180
 1020 CL(J)=0.                                                          OPTM1190
      DO 1050 J=1,NN                                                    OPTM1200
      READ (NTAPEX) (A(I),I=1,NN)                                       OPTM1210
      IF (J-NW)1050,1050,1030                                           OPTM1220
 1030 CONTINUE                                                          OPTM1230
      DO 1040 I=1,NW                                                    OPTM1240
 1040 CL(I)=CL(I)+A(I)*B(J)                                             OPTM1250
 1050 CONTINUE                                                          OPTM1260
C                                                                       OPTM1270
C ------------------                                                    OPTM1280
      RETURN                                                            OPTM1290
C ------------------                                                    OPTM1300
      END                                                                      -
CDECK DCPD
      SUBROUTINE DCPD(NM,NTAPEX,A,ALPHAM,CLM)                           DCPD1000
C                                                                       DCPD1010
C ..................................................................... DCPD1020
C     COMPUTES WING PANEL PRESSURE DIFFERENCE                           DCPD1030
C ..................................................................... DCPD1040
C                                                                       DCPD1050
      DIMENSION A(1),ALPHAM(1),CLM(1)                                   DCPD1060
C                                                                       DCPD1070
      DO 1000 J=1,NM                                                    DCPD1080
      CLM(J)=0.0                                                        DCPD1090
 1000 CONTINUE                                                          DCPD1100
C                                                                       DCPD1110
      DO 1010 J=1,NM                                                    DCPD1120
      READ (NTAPEX) (A(I),I=1,NM)                                       DCPD1130
      DO 1010 I=1,NM                                                    DCPD1140
      CLM(I)=CLM(I)+A(I)*ALPHAM(J)                                      DCPD1150
 1010 CONTINUE                                                          DCPD1160
C                                                                       DCPD1170
C ------------------                                                    DCPD1180
```

```
      RETURN                                                            DCPD1190
C ------------------                                                    DCPD1200
      END                                                               -
CDECK CAMBWB
      SUBROUTINE CAMBWB (NB,NW,NTAPEX,NTAPEY,A,CLW,ALPHAB,ALPHAW)       CAMB1000
C                                                                       CAMB1010
C ..................................................................... CAMB1020
C     FOR WING-BODY CASES, COMPUTES NORMAL VELOCITY COMPONENTS ON WING  CAMB1030
C ..................................................................... CAMB1040
C                                                                       CAMB1050
      DIMENSION A(1),ALPHAW(1),ALPHAB(1),CLW(1)                         CAMB1060
C                                                                       CAMB1070
      DO 1000 J=1,NW                                                    CAMB1080
      ALPHAW(J)=0.                                                      CAMB1090
 1000 CONTINUE                                                          CAMB1100
      DO 1010 J=1,NW                                                    CAMB1110
      READ (NTAPEX) (A(I),I=1,NW)                                       CAMB1120
      DO 1010 I=1,NW                                                    CAMB1130
      ALPHAW(I)=ALPHAW(I)+A(I)*CLW(J)                                   CAMB1140
 1010 CONTINUE                                                          CAMB1150
      DO 1020 J=1,NB                                                    CAMB1160
      READ (NTAPEY) (A(I),I=1,NW)                                       CAMB1170
      DO 1020 I=1,NW                                                    CAMB1180
      ALPHAW(I)=ALPHAW(I)+A(I)*ALPHAB(J)                                CAMB1190
 1020 CONTINUE                                                          CAMB1200
C                                                                       CAMB1210
C ------------------                                                    CAMB1220
      RETURN                                                            CAMB1230
C ------------------                                                    CAMB1240
      END                                                               -
CDECK DCPI
      SUBROUTINE DCPI (NB,NW,NTAPEX,A,ALPHAB,CLW,CLB)                   DCPI1000
C                                                                       DCPI1010
C ..................................................................... DCPI1020
C     COMPUTES BODY PANEL PRESSURE DIFFERENCE                           DCPI1030
C ..................................................................... DCPI1040
C                                                                       DCPI1050
      DIMENSION A(1),ALPHAB(1),CLW(1),CLB(1)                            DCPI1060
C                                                                       DCPI1070
      DO 1000 J=1,NB                                                    DCPI1080
      CLB(J)=0.                                                         DCPI1090
 1000 CONTINUE                                                          DCPI1100
      DO 1010 J=1,NB                                                    DCPI1110
      READ (NTAPEX) (A(I),I=1,NB)                                       DCPI1120
      DO 1010 I=1,NB                                                    DCPI1130
      CLB(I)=CLB(I)+A(I)*ALPHAB(J)                                      DCPI1140
 1010 CONTINUE                                                          DCPI1150
      DO 1020 J=1,NW                                                    DCPI1160
      READ (NTAPEX) (A(I),I=1,NB)                                       DCPI1170
      DO 1020 I=1,NB                                                    DCPI1180
      CLB(I)=CLB(I)+A(I)*CLW(J)                                         DCPI1190
 1020 CONTINUE                                                          DCPI1200
C                                                                       DCPI1210
C ------------------                                                    DCPI1220
      RETURN                                                            DCPI1230
C ------------------                                                    DCPI1240
      END                                                               -
      SUBROUTINE CAMBW(NW,NTAPEX,A,   CLW,ALPHAW)                       CAMB1000
C                                                                       CAMB1010
C ..................................................................... CAMB1020
C     FOR WING ONLY CASE, COMPUTES NORMAL VELOCITY COMPONENTS ON WING   CAMB1030
C ..................................................................... CAMB1040
C                                                                       CAMB1050
      DIMENSION A(1),ALPHAW(1),CLW(1)                                   CAMB1060
C                                                                       CAMB1070
```

```
      DO 1000 J=1,NW                                                    CAMB1080
      ALPHAW(J)=0.                                                      CAMB1090
 1000 CONTINUE                                                          CAMB1100
      DO 1010 J=1,NW                                                    CAMB1110
      READ (NTAPEX) (A(I),I=1,NW)                                       CAMB1120
      DO 1010 I=1,NW                                                    CAMB1130
      ALPHAW(I)=ALPHAW(I)+A(I)*CLW(J)                                   CAMB1140
 1010 CONTINUE                                                          CAMB1150
C                                                                       CAMB1160
C ------------------                                                    CAMB1170
      RETURN                                                            CAMB1180
C ------------------                                                    CAMB1190
      END                                                                      -
CDECK PBODY
      OVERLAY(WANG, 7, 10)
      PROGRAM  PBODY
C
C
C              READ INPUT OF FINITE ELEMENT GRID SPECIFICATIONS FROM
C            BING  OUTPUT.
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1         NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1         ,XC (3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH
     2         ,CPB, SID(20)
      COMMON/REPEAT/ IREPT
      DIMENSION CPB(110)
      DIMENSION XS(115,7),YS(115,7),CPBS(115,7)
      DIMENSION XI(3,350), PB(700)
C
      DIMENSION SUM(10)
C
C
C
C        CHECK SIGN OF SID(NSYM) FOR PRESSURE OUTPUT SIGN AS DESIRED.
C
      ASID = ABS( SID(NSYM) )
      PSIGN = SID(NSYM)/ ASID
      NSID = IFIX( ASID )
      PCONV = 0.5 * 1.4 * PINF * XMACH**2
      IREPT = 0
      IF ( NPOLAR  .GT. 0 ) GO TO 300
C
C        READ IN NUMBER OF GRID ( NGRIDP) AND ELEMENT ( NE ) CARDS.
C
      READ (5,901)  NGRIDP, NE
C
C              CONVERT GRID COORDINATE TO RECTANULAR COORDINATE XI ( X,
C          Y, Z )
C
      NS = NSYM
      CALL CONVT( XI, NG, NGRIDP, NS,  ISYM)
C
C              TRANSFORM GRID COORDINATE TO THE AERO CODE COORDINATE
C          SYSTEM  XI( X, Y, Z ).
C
      CALL COORDT( XI, NGRIDP )
C
C              COMPUTE THE CENTROID OF ELEMENT XC( X, Y, Z ) .
C
      CALL CENTRO( XI, XC, NG, NGRIDP, NELM, NE)
C
C              EXPRESS CENTROID XC( X, Y, Z ) IN TERMS OF SURFACE
```

```
C         PARAMETERS  X  AND  THETA    FOR SURFACE FIT.
C
      CALL CONVBK( XC, NE)
C
C             STORE XMIDB, THETB  AND  CPBB  INTO LOCATIONS OF ONE
C        DIMENSIONAL ARRAY FOR SURFACE FIT.
C
C
C             FOR REDUCING COMPUTING TIME OF INVERTING A LARGE MATRIX,
C             THE INTERPOLATION IS DIVIDED INTO SEVERAL PATCHES OF
C             EQUAL OR LESS THAN 100 POINTS.
C
  300 CONTINUE
      NSEG = 2
      NTS = NTHETB / NSEG + 1
      NPS = NBODYS * NTS
      N3 = NPS + 3
      DO 400 ISEG = 1,NSEG
      DO 500 J = 1, NTS
      JT = J + (ISEG-1) * (NTS-1)
      DO 500 I=1,NBODYS
      IJ = I + (J-1)*NBODYS
      XS(IJ, ISEG) = XMIDB(I)
      YS(IJ,ISEG) = THETB(JT)
      CPBS(IJ, ISEG) = CPBB(I, JT)
  500 CONTINUE
      CALL SURFIT( XS( 1,ISEG), YS( 1,ISEG), N3, CPBS( 1,ISEG ) )
  400 CONTINUE
C
      DO 600 I=1,NE
C
C             GROUP THE FINITE ELEMENT CENTROID LOCATION  XC  IN THEIR
C            PROPER SEGMENT.
C
      CORANG = XC( 2, I)
      IF ( CORANG  .GT. 180.0 ) CORANG = 360.0 - CORANG
      JT = 1 + NBODYS*(NTS - 1)
      DO 605  K=1,NSEG
      NSG  = K
      IF ( CORANG  .LT. YS( JT, K) ) GO TO 608
  605 CONTINUE
  608 CONTINUE
C
      SUM(NSG) =CPBS(1,NSG) +CPBS(2,NSG) * XC(1,I) +CPBS(3,NSG)*CORANG
      DO 610  J = 1, NPS
      J3 = J+3
      RI = (   XC(1,I) -XS(J,NSG) )**2 + ( CORANG  -  YS(J,NSG) )**2
      IF ( RI  .LE. 1.E-9 ) GO TO 610
      SUM(NSG) = SUM(NSG) +CPBS(J3,NSG) * RI * ALOG( RI )
  610 CONTINUE
      PB(I) = SUM(NSG) * PCONV * PSIGN
C
C             PUNCH PRESSURE AND GRID POINT IDENTIFICATION IN THE FORM
C         OF BULK DATA FORMAT OF NASTRAN.
C
      IF ( IPUNCH  .EQ. 0 ) GO TO 600
      PUNCH  902, (NSID, PB(I), NELM(I)  )
  600 CONTINUE
C
      WRITE (6,903) ( NELM(I),   XC(1,I),   XC(2,I), PB(I), I=1,NE)
  902 FORMAT (8HPLOAD2  ,I8,F8.2,I8)
  903 FORMAT (1H1//*      AERODYNAMIC PRESSURE ON THE BODY - - CENTROID *
     1              ,*OF FINITE ELEMENT AS GENERATED BY NSWC CODE IN THE*/
     A        7X,*SAME LOCAL COORDINATE SYSTEM*///3X,
     2             *ELEMENT*,6X,*X*,12X,*THETA*,9X,*PRESSURE*,13X,*ELEMENT*,
```

```
     3          6X,*X*,12X,*THETA*,9X,*PRESSURE*//(1H0,2(I8,E12.3,2X,
     4          E12.3,4X,E12.3,10X) ) )
  901 FORMAT (2I10)
C
C
      END
CDECK CONVBK
      SUBROUTINE CONVBK( X, NE)
C
C
C              CONVERT COORDINATE X, Y, Z INTO X AND THETA FOR SURFACE
C           FIT PURPOSES.
C
      DIMENSION X( 3, NE)
C
      RD = 57.2957795
C
      DO 100 I = 1, NE
      R = SQRT( X(2, I) **2 + X( 3, I)**2 )
      ANG = ACOS ( X( 3, I) / R ) * RD
      IF ( X( 2, I)  .GT.  0.0 ) X( 2, I) = ANG
      IF ( X( 2, I)  .LT.  0.0 ) X( 2, I) = 360.0 - ANG
  100 CONTINUE
C
C
      RETURN
      END
CDECK PSUM
      OVERLAY(WANG, 7, 11)
      PROGRAM PSUM
C
C
C           THIS SUBROUTINE INTERPOLATES THE PRESSURE COEFFICIENTS CPB (
C           EFFECT OF WING ON THE BODY ) AT THE BODY STATIONS AND ADDS TO
C           THAT OF THE BODY ALONE CPBB.
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1          SYM, KACE, NPOLAR, IRW
      COMMON/ NVAR / NFMT(9), NROW(2), NS, NPANEL, NACEL, NROWB, NROWW,
     1          NCOLW, NTHETA, NTHETS, NXLE, NRG, NPOLR, NCLX
      COMMON /TVAR/  TITLE(20),THETA(210),THETAB(11),THETAS(11),THETAN(
     1             11,2),TCL(11),TCD(11),TCM(11), THICK, TWIST
      COMMON /XVAR/  XBAR(210),XC(210),XNI(2),XNXN(2),XNTN(2),XB(51),XN(
     1              51,2),XNNCD(2),XNNCL(2),XNNCM(2),XYZ(3),XCL(11),XCD
     2              (11), XNACEL,XP,XCPBAR
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1 ,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH, CPB
      COMMON/REPEAT/ IREPT
      DIMENSION XADD(115), YADD(115), CF(115)
      DIMENSION CPB(110)
C
      IREPT = 0
      CDR = 57.2957795
      NTHETA = NBODY / NROWB
      DO 100 I=1,NTHETA
      DO 90 J=1,NROWB
      IJ = J + (I-1)* NROWB
      XADD(IJ) = XC(J)
      YADD(IJ) = THETAB(I)
      CF(IJ) = CPB(IJ)
   90 CONTINUE
  100 CONTINUE
C
      NB3 = NBODY + 3
      CALL SURFIT( XADD, YADD, NB3, CF)
```

```
C
      DO 110 I=1,NBODYS
      DO 110 J =1,NTHETB
      IF ( XMIDB(I) .LT. XADD(1)  .OR. XMIDB(I) .GT. XADD(NBODY) )GO TO
     1           110
      SUMB = CF(1) + CF(2) * XMIDB(I) + CF(3) * THETB(J)
      DO 120 K=1,NBODY
      K3 = K+3
      RI = ( XMIDB(I) - XADD(K) )**2 + ( THETB(J) - YADD(K) )**2
      IF ( RI .LT. 1.E-9 ) GO TO 120
      SUMB = SUMB + CF(K3) * RI * ALOG( RI )
  120 CONTINUE
      CPBB(I,J) = CPBB(I,J) + SUMB
  110 CONTINUE
C
      WRITE (6,1830)
      WRITE (6,1860) ( THETB(I), I=1,NTHETB)
      WRITE (6,1870)
      NBS = NBODYS
      IF ( XMACH  .LT. 1.0 ) NBS = NBODYS - 1
      DO 200 I=1,NBS
      WRITE (6,1880) XMIDB(I), ( CPBB(I,J), J=1,NTHETB)
  200 CONTINUE
C
C
 1830 FORMAT (1H0/2X,*BODY PRESSURE COEFFICIENTS (CP), INCLUDING THE EF*
     1           ,*FECT OF WING*)
C
 1860 FORMAT (1H0,4X,*THETA (DEG)*,F14.4,9F10.4/(F29.4,9F10.4) )
C
 1870 FORMAT (1H ,9X,*X*)
C
 1880 FORMAT (1H ,F14.4,F15.5,9F10.5/(F29.5,9F10.5) )
C
      END
CDECK WINGPR
      OVERLAY(WANG, 7, 12)
      PROGRAM WINGPR
C
C
C        READ IN FINITE ELEMENT , INTERPOLATE PRESSURE AND GENERATES
C        PRESSURE LOAD ON THE WING.
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1           SYM, KACE, NPOLAR, IRW
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/NVAR / NFMT(9), NROW(2), NS, NPANEL, NACEL, NROWB, NROWW,
     1           NCOLW, NTHETA, NTHETS, NXLE, NRG, NPOLR, NCLX
      COMMON/PBDATA/ NBODYS,XMIDB(51), NTHETB,THETB(11),CPBB(51,11)
     1           , XCB(3,700), NG(350), NELM(700), NE,NGRIDP,PINF,IPUNCH
     2           , CPB
      COMMON/WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
      COMMON /XVAR/  XBAR(210),XC(210),XNI(2),XNXN(2),XNTN(2),XB(51),XN(
     1               51,2),XNNCD(2),XNNCL(2),XNNCM(2),XYZ(3),XCL(11),XCD
     2               (11), XNACEL,XP,XCPBAR
      COMMON/YVAR/ YEAR(210), YC(210), YNI(2)
      COMMON/ZVAR/ ZEAR(210), ZC(210),ZNI(2),ZDELTA(51),ZDN(51,2),ZP,ZA
      DIMENSION CPU(110), CPL( 110)
      DIMENSION CPB(110)
C
      CALL WINGO( XC, YC)
C
      END
CDECK WINGO
      SUBROUTINE WINGO( XC, YC)
```

```
C
C        INTERPOLATE OUTPUT FOR NASTRAN USE, GRID POINTS AND ELEMENT
C        LOCATIONS ARE GENERATED FROM PING.
C
      COMMON/ MAIN / LA, LB, LC, LD, LE, LF, LI,LO, NBODY, NWING, XMACH,
     1          SYM, KACE, NPOLAR, IRW
      COMMON/COEFFS/ CF(100, 4, 2)
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/NVAR / NFMT(9), NROW(2), NS, NPANEL, NACEL, NROWB, NROWW,
     1          NCOLW, NTHETA, NTHETS, NXLE, NRG, NPOLR, NCLX
      COMMON/PCOORD/ X(210), Y(210), NPT
      COMMON/WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
      COMMON/PEDATA/ NBODYS,XMIDB(51), NTHETB,THETB(11),CPBB(51,11)
     1          , XCB(3,700), NG(350), NELM(700), NE,NGRIDP,PINF,IPUNCH
     2          , CPB
      COMMON/REPEAT/ IREPT
      DIMENSION CPB(110)
      DIMENSION CPU(110), CPL(110), XC(1), YC(1)
      DIMENSION NELMT( 6)
      DIMENSION COF(115)
C
      PCONV = 0.5 * 1.4 * PINF * XMACH * XMACH
      IREPT = 0
C
C
      NW = 1
      NPT = ( NPANEL - NBODY ) / NWG
   10 CONTINUE
C
C        STORE  PRESSURE AT CONTROL POINT AND ITS LOCATION FOR
C        INTERPOLATION.
C
      NC1 = NPT
      NC3 = NC1 + 3
C
      IF ( ISOLID  .EQ. 0 ) GO TO 205
C
C       SOLID WING CASE , LIFTING PRESSURE
C
      DO 200 J=1,NC1
      JA = J + ( NW-1)*NPT
      COF (J) = ( CPL(JA) - CPU(JA) ) * PCONV
      JB = JA + NBODY
      X(J) = XC(JB)
      Y(J) = YC(JB)
  200 CONTINUE
C
      WRITE (6,803) ( X(I), Y(I), COF(I), I=1,NC1)
C
      CALL SURFIT( X, Y, NC3, COF)
C
      DO 201 K=1,NC3
      CF( K,NW, 1) = COF(K)
  201 CONTINUE
      GO TO 250
C
  205 CONTINUE
      DO 210  J =1, NC1
      JA = J + ( NW-1)*NPT
      COF(J) = CPU(JA) * PCONV
      JB = JA + NBODY
      X(J) = XC(JB)
      Y(J) = YC(JB)
  210 CONTINUE
C
```

```
      WRITE (6,801)  ( X(I), Y(I), COF(I), I=1, NC1 )
C
      CALL SURFIT( X, Y, NC3, COF)
C
      DO 230 K=1, NC3
      CF( K,NW, 1) = COF( K )
  230 CONTINUE
C
      DO 215  J =1, NC1
      JA = J + ( NW-1)*NPT
      COF(J) = CPL(JA) * PCONV
  215 CONTINUE
C
      WRITE (6,802)  ( X(I), Y(I), COF(I), I=1, NC1 )
C
      CALL SURFIT( X, Y, NC3, COF)
C
      DO 240 K=1, NC3
      CF( K,NW, 2) = COF( K )
  240 CONTINUE
C
  250 CONTINUE
C
C          COMPUTES PRESSURE AT DESIRED LOCATION ( CENTROID OF ELEMENT ),
C          FOR EACH COMPLETE WING.
C
      CALL PWING( NW )
C
      NW = NW + 1
C
C
      IF ( NW  .LE.  NWG ) GO TO 10
C
  801 FORMAT (1H0,* PRESSURE AT CONTROL POINT - UPPER SURFACE*/ 15X,
     1           *X*,9X,*Y*,9X,*PRESSURE*///(11X,F8.4,2X,F8.4,2X,E12.4) )
  802 FORMAT (1H0,* PRESSURE AT CONTROL POINT - LOWER SURFACE*/ 15X,
     1           *X*,9X,*Y*,9X,*PRESSURE*///(11X,F8.4,2X,F8.4,2X,E12.4) )
  803 FORMAT (1H1,*      TOTAL LIFTING PRESSURE ON THE WING AT THE*
     A           * CONTROL POINT*//15X,*X*,
     1           9X,*Y*,9X,*PRESSURE*///(11X,F8.4,2X,F8.4,2X,E12.4) )
C
      RETURN
      END
CDECK PWING
      SUBROUTINE PWING( NW )
C
C          INTERPOLATES PRESSURE ON WING AT GIVEN LOCATION X,Y FOR
C          LOCATION XC, YC OR CENTROID OF ELEMENT
C
      COMMON/COEFFB/ CF(100, 4, 2)
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH, CPB
     2           , SID(20)
      COMMON/PCOORD/ X(210), Y(210), NP
      COMMON/ WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
      DIMENSION CPU(110), CPL(110)
      DIMENSION P(210)
      DIMENSION CPB(110), XI(3,350)
      INTEGER CHECK, WORD1
C
      DATA  WORD1/ 5HCQUAD/
C
C         WRITE TITLE OF WING PRESSURE OUTPUT
C
```

```
      IF ( NW  .GT. 1 ) GO TO 50
      WRITE (6,901)
      WRITE (6,908)  NWG
   50 CONTINUE
      WRITE (6,910)  NW
      NPUNCH = 0
      NSID = IFIX( SID(NSYM) )
      NS = 2
      IF ( ISOLID  .NE. 0 ) NS = 1
      DO 150 K=1,NS
C
C        CHECK SIGN OF SID(NSYM) FOR PRESSURE OUTPUT SIGN AS DESIRED.
C
      ASID = ABS( SID(NSYM) )
      PSIGN = SID(NSYM)/ ASID
      NSID = IFIX( ASID )
C
C
C              READ IN FINITE ELEMENT SPECIFICATIONS FROM PING OUTPUT
C
C
C         READ IN NUMBER OF GRID (NG) AND ELEMENT SPECIFICATION (NE)
C         CARDS.
C
      READ (5,902)  NGRIDP,  NE
C
      CALL CONVT( XI, NG, NGRIDP, NSYM, ISYM)
C
      CALL COORDT( XI, NGRIDP)
C
      CALL CENTRO( XI, XCB,NG, NGRIDP, NELM, NE)
C
      DO 100 IE = 1,NE
C
C        CHANGE THE SIGN OF Y COORDINATE OF THE LEFT WING TO REFLECT
C        THE SYMMETRY OF WING ARRANGEMENT
C
      YCOORD = ABS( XCB( 2, IE ) )
C
      SUM = CF(1,NW,K) + CF(2,NW,K)*XCB(1,IE) + CF(3,NW,K)*YCOORD
      DO 110 J = 1, NP
      J3 = J + 3
      RI = ( XCB(1,IE) - X(J) )**2 + ( YCOORD - Y(J) )**2
      IF ( RI  .LT. 1.E-9 ) GO TO 110
      SUM = SUM + CF( J3, NW, K )* RI * ALOG( RI )
  110 CONTINUE
      P(IE) = SUM * PSIGN
  100 CONTINUE
      IF (IFORM(NW) .NE. 1 ) NSYM =NSYM
      IF (IFORM(NW) .EQ. 1 ) NSYM =NSYM + 1
C
      IF ( ISOLID  .EQ. 0  .AND.  K  .EQ. 1 ) WRITE (6,903)
      IF ( ISOLID  .EQ. 0  .AND.  K  .EQ. 2 ) WRITE (6,904)
C
      WRITE (6,909)
      WRITE (6,906) ( NELM(J), XCB(1,J), XCB(2,J), P(J), J=1,NE )
C
C         PUNCH  PRESSURE AND GRID POINT IDENTIFICAITON IN THE FORM OF
C         BULK DATA FORMAT OF NASTRAN
C
      IF ( IPUNCH  .EQ.  0 ) GO TO 150
      DO 220  L=1, NE
      PUNCH 807, ( NSID, P(L), NELM(L) )
  220 CONTINUE
      NPUNCH = NPUNCH + NE
```

```
  150 CONTINUE
C         WRITE NUMBER OF BULK DATA CARDS PUNCHED
C
      IF ( IPUNCH  .EQ.  1 ) WRITE (6,905) NPUNCH
C
C
C
C
  807 FORMAT ( 8HPLOAD2  ,I8,F8.2,I8 )
  901 FORMAT (1H1//*      AERODYNAMIC PRESSURE ON THE WING - - CENTROID *
     1        ,*OF FINITE ELEMENT AS GENERATED BY NSWC CODE IN THE*/
     A        7X,*SAME LOCAL COORDINATE SYSTEM*/)
  902 FORMAT  (2I10 )
  903 FORMAT (1H0,10X,*PRESSURE LOAD ON THE UPPER SURFACE OF WING*//)
  904 FORMAT (///1H0,10X,*PRESSURE LOAD ON THE LOWER SURFACE OF WING*//)
  906 FORMAT (1H0,2(I7,E13.3,3X,E13.3,4X,E13.3,6X) /(1X,2(I7,E13.3,3X,
     1        E13.3,4X,E13.3,6X)  ) )
  907 FORMAT (1H0,10X,*SOLID WING, SUM OF UPPER AND LOWER SURFACE*,
     1        * PRESSURES*//)
  908 FORMAT (1H0,10X,*TOTAL NUMBER OF WINGS  *,I5/)
  909 FORMAT (1H0/3X,*ELEMENT*,9X,*X*,11X,*Y*,11X,*PRESSURE*,13X,
     1        *ELEMENT*,9X,*X*,11X,*Y*,11X,*PRESSURE*//)
  910 FORMAT (1H0,10X,*WING  NO. *,I5/ )
C
  905 FORMAT (1H ///*         NUMBER  OF  CARD  PUNCHED  FOR  THIS  RUN
     1        *, I10 /)
C
      RETURN
      END
CDECK SAVE
      OVERLAY(WANG, 7, 13)
      PROGRAM SAVE
C
C
      COMMON/ MAIN / NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1        NTAPEO,NBODY,NWING,XMACH,SYM,KACE,NPOLAR,IRW
      COMMON / BODYSP/ DRDX(51), DZDX(51)
      COMMON /AVAR/  A(210),ACB(21),ABX(100),AWX(110),AREA(210),ARN(2),A
     1               RWT(20),ALPHA(210),ALPHAB(210),ALPHAS(210),ALPHAT(1
     2               10),ALPHAW(110),ALPHAX(110),AWS(110),ABT(100),AN1N(
     3               210,2),ALPHAA,ALPHAD,ARA,ARADEG,ARB,ARW,ARAS,AT,AAA
      COMMON /BODS/ NUMBOD,XNOS(3),YNOS(3),ZNOS(3),NBSTAT(3)
      COMMON /BVAR/  B(210) , BBCL, BBCD, BBCM
      COMMON /CVAR/  C(210),CHORD(210),CL(210),CPNN(51,11,2),CLS(210)
     1               ,CDR,CASE,CPCALC,CBAR,CONSNT,CLBAR,CLX,CLM,CDM
      COMMON /DVAR/  D(210),DZDXB(51), DADEG, DARAD
      COMMON /IVAR/  IPOLAR
      COMMON /KVAR/  KASE,KONFIG,KPOLAR
      COMMON /NVAR/  NFMT(9),NROW(2)
     1               ,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,NTHETS,
     2                   NXLE,NRG,NPOLR ,NCLX
      COMMON /PVAR/  POLAR
      COMMON /RVAR/  R(51),RN(51,2), RATIOX, RFAREA
      COMMON /SVAR/  SEMIS,SLC
      COMMON /TVAR/  TITLE(20),THETA(210),THETAB(11),THETAS(11),THETAN(
     1             11,2),TCL(11),TCD(11),TCM(11),THICK,TWIST
      COMMON /UVAR/  UBWT(100),UWBT(110),UWWT(110),UNCL(210,2)
      COMMON /VVAR/  VBWT(100),VWBT(110),VWWT(110),VNCL(210,2)
     1               ,VOUT
      COMMON /WVAR/  WBWT(100),WWBT(110),WWWT(110),WNCL(210,2)
      COMMON /XVAR/  XBAR(210),XC(210),XNI(2),XNXN(2),XNTN(2),XB(51),XN(
     1               51,2),XNNCD(2),XNNCL(2),XNNCM(2),XYZ(3),XCL(11),XCD
     2               (11), XNACEL, XP, XCPBAR
      COMMON /YVAR/  YBAR(210),YC(210),YNI(2)
      COMMON /ZVAR/  ZBAR(210),ZC(210),ZNI(2),ZDELTA(51),ZDN(51,2)
```

```
     1                 ,ZP,ZA
      COMMON/PBDATA/ NBODYS, XMIDB(51), NTHETB, THETB(11), CPBB(51,11)
     1,XCB(3,700),NG(350),NELM(700),NE,NGRIDP,PINF,IPUNCH, CPB
      COMMON/CORTRN/ NSYM,NTRANS(20),ISYM(20),VA(3,2,20), BTFM(9,2,20)
      COMMON/WINGOT/ CPU, CPL, NWG, ISOLID,IFORM(10)
      COMMON/EXTRA/ ARAN, DSB1, DSB2, DS1, DS2, NB1,NB2,NS1,NS2,
     1          NTN, NXN, XNOSE
C
      DIMENSION CPU(110), CPL( 110),CPB(110),XNS1(2),XNS2(2),NS1(2),
     1          NS2(2), DS1(2), DS2(2)
C
      REWIND 12
      IF ( IRW  .EQ. 2 ) GO TO 100
C
      WRITE(12) DATE,NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1      NTAPEO,NBODY,NWING,XMACH, SYM,KACE, DRDX, DZDX, NPOLAR, A,
     2          ACB, ABX, AWX, AREA, ARN,ARWT, ALPHA, ALPHAB, ALPHAS,
     3          ALPHAT,ALPHAW, ALPHAX, AWS,ABT,AN1N,ALPHAA,ALPHAD,ARA,
     4          ARADEG,ARB,ARW,ARAS,AT,AAA,NUMBOD,XNOS,YNOS,ZNOS,NBSTAT,
     5    B,BBCL,BBCD,BBCM,C,CHORD,CL,CPNN,CLS,CDR,CASE,CPCALC,CBAR,
     6    CONSNT,CLBAR,CLX,CLM,CDM,D,DZDXB,DADEG,DARAD,IPOLAR,KASE,KONFIG
     7    ,KPOLAR,NFMT,NROW,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,
     8    NTHETS,NXLE,NRG,NPOLR,NCLX,POLAR,R,RN,RATIOX,RFAREA,SEMIS,SLC,
     9    TITLE,THETA,THETAB,THETAS,THETAN,TCL,TCD,TCM,THICK,TWIST,UBWT,
     A    UWBT,UWWT,UNCL,VBWT,VWBT,VWWT,VNCL,VOUT,WBWT,WWBT,WWWT,WNCL,
     B    XBAR,XC,XNI,XNXN,XNTN,XB,XN,XNNCD,XNNCL,XNNCM,XYZ,XCL,XCD,XP,
     C    XNACEL,XCPBAR,YBAR,YC,YNI,ZBAR,ZC,ZNI,ZDELTA,ZDN,ZP,ZA,NBODYS,
     D    XMIDB,NTHETB,THETB,CPBB,XCB,NG,NELM,NE,NGRIDP,PINF,IPUNCH,CPB,
     E    NSYM,NTRANS,ISYM,VA,BTFM,CPU,CPL,NWG,ISOLID,ARAN,DSB1,DSB2,DS1,
     F    DS2,NB1,NB2,NS1,NS2,NTN,NXN,XNOSE
C
      GO TO 200
C
  100 CONTINUE
C
      READ (12) DATE,NTAPEA,NTAPEB,NTAPEC,NTAPED,NTAPEE,NTAPEF,NTAPEI,
     1      NTAPEO,NBODY,NWING,XMACH, SYM,KACE, DRDX, DZDX, NPOLAR, A,
     2          ACB, ABX, AWX, AREA, ARN,ARWT, ALPHA, ALPHAB, ALPHAS,
     3          ALPHAT,ALPHAW, ALPHAX, AWS,ABT,AN1N,ALPHAA,ALPHAD,ARA,
     4          ARADEG,ARB,ARW,ARAS,AT,AAA,NUMBOD,XNOS,YNOS,ZNOS,NBSTAT,
     5    B,BBCL,BBCD,BBCM,C,CHORD,CL,CPNN,CLS,CDR,CASE,CPCALC,CBAR,
     6    CONSNT,CLBAR,CLX,CLM,CDM,D,DZDXB,DADEG,DARAD,IPOLAR,KASE,KONFIG
     7    ,KPOLAR,NFMT,NROW,NS,NPANEL,NACEL,NROWB,NROWW,NCOLW,NTHETA,
     8    NTHETS,NXLE,NRG,NPOLR,NCLX,POLAR,R,RN,RATIOX,RFAREA,SEMIS,SLC,
     9    TITLE,THETA,THETAB,THETAS,THETAN,TCL,TCD,TCM,THICK,TWIST,UBWT,
     A    UWBT,UWWT,UNCL,VBWT,VWBT,VWWT,VNCL,VOUT,WBWT,WWBT,WWWT,WNCL,
     B    XBAR,XC,XNI,XNXN,XNTN,XB,XN,XNNCD,XNNCL,XNNCM,XYZ,XCL,XCD,XP,
     C    XNACEL,XCPBAR,YBAR,YC,YNI,ZBAR,ZC,ZNI,ZDELTA,ZDN,ZP,ZA,NBODYS,
     D    XMIDB,NTHETB,THETB,CPBB,XCB,NG,NELM,NE,NGRIDP,PINF,IPUNCH,CPB,
     E    NSYM,NTRANS,ISYM,VA,BTFM,CPU,CPL,NWG,ISOLID,ARAN,DSB1,DSB2,DS1,
     F    DS2,NB1,NB2,NS1,NS2,NTN,NXN,XNOSE
C
  200 CONTINUE
      REWIND 12
C
      END
```

DISTRIBUTION LIST

	Copies
Navy	
Naval Surface Weapons Center White Oak Laboratory Silver Spring, Maryland 20910	
WA-30	1
WA-30 (Huang)	30
WA-31	1
WA-31 (Matra)	10
WA-31 (Kiddy)	10
WA-31 (Graff)	1
WA-11 (Hardy)	1
WA-43 (Edwards)	1
WR-15 (Kushner)	1
WX-21	3
WS-10 (Reinhard)	1
Naval Surface Weapons Center Dahlgren Laboratory Dahlgren, Virginia 22448	
DK-74 (Blackman)	1
DG (Soper)	1
Office of Chief of Naval Operations Operations Evaluation Group (OP-03EG) Washington, D. C. 20350	1
Defense Nuclear Agency Washington, D. C. 20301	2
Defense Documentation Center Cameron Station Alexandria, Virginia 22314	12
Commander Naval Sea Systems Command Washington, D. C. 20360	
SEA-03A	1
SEA-0333	1
SEA-035-13	2
SEA-035A	1
SEA-55212	1

	Copies
SEA-0632	2
SEA-654312	1
PMS-403-02	1
Commander Naval Air Systems Command (AIR-320-B) Washington, D.C. 20360	1
Commander Pacific Missile Test Center Point Mugu, California 93042	
T. W. Elliott (Code 5331)	1
Commander Naval Air Development Center Johnsville, Warminister, Pennsylvania 18974	
Dr. E. McQuillan (Code 3033)	1
Herman Rubin (Code 3033)	1
K. T. Yen (Code 30331)	1
Naval Research Laboratory Washington, D.C. 20390	
Dr. M. R. Achter (Code 6340)	1
Commander Naval Weapons Center China Lake, California 93557	
Code 4062	1
Code 45703	1
Mr. E. L. Jeter	1
James E. Serpanos	1
Office of Naval Research Department of the Navy Washington, D.C. 20360	
Code 461	1
Code 439 (N. Perrone)	1
Code 470	1
Technical Library	2
Naval Postgraduate School Monterey, California 93940	
Code 2124	1
Dr. J. G. Cantin	1
Commanding Officer Naval Ship Research and Development Center Bethesda, Maryland 20034	
Code 1844 (Jim McKee)	1
Code 1844 (Gordon Everstine)	1
Code 564	2

	Copies
Air Force	
AFFDL/PTS (CAPT Gilbert L. Camburn) AF Flight Dynamics Laboratory Wright-Patterson Air Force Base, Ohio 45433	1
AF Materials Laboratory Wright-Patterson Air Force Base, Ohio 45433 Code MAAN (B. R. Emrich)	1
Flight Dynamics Laboratory Structures Division Wright-Patterson Air Force Base, Ohio 45433 Mr. Leo J. D. Bernier	1
AFML/LLM (W. G. Ramke) Wright Patterson Air Force Base, Ohio 45433	1
AFATL/DLRV Eglin Air Force Base, Florida 32542 James M. Heard	2
AFAL/TEM-3 Wright-Patterson Air Force Base, Ohio 45433 Richard A. Ireland	1
Army	
Commanding General U.S. Army Missile Command Redstone Arsenal, Alabama 35809	1
Other Agencies	
NASA Scientific and Technical Information Facility P.O. Box 33 College Park, Maryland 20740 Acquisitions Branch - Foreign Exchange	2
Industry	
Aerospace Corporation 1111 East Mill Street San Bernardino, California 92408 Dr. Robert L. Hallse	1
Booz Allen Applied Research P.O. Box 874 Shalimar, Florida 32579 William Day	1

	Copies
Applied Physics Laboratory The Johns Hopkins University 8621 Georgia Avenue Silver Spring, Maryland 20910	
L. B. Weckesser	1
W. C. Caywood	1
R. M. Rivello	1
Battelle Memorial Institute Defense Metals Information Center 505 King Avenue Columbus, Ohio 43201	
H. D. Moran	1
Brunswick Corporation Technical Products Division 325 Brunswick Lane Marion, Virginia 24354	
Robert Copeland	1
General Dynamics/Pomona P.O. Box 2507 Pomona, California 91766	
R. A. Miller	1
D. A. Underhill	1
University of Maryland College Park, Maryland 20740	
Dr. Schaeffer	1
Raytheon Company Missile Systems Division, VB1-3 Hartwell Road Bedford, Massachusetts 01730	
R. D. Howe	1
D. H. Sanders	1
McDonnell Douglas Astronautics Co. 5301 Bolsa Avenue Huntington Beach, California 92647	
W. F. Bozich	1
K. K. Wang	1
R. S. Lee	1
McDonnell Douglas Astronautics Co. Eastern Division P.O. Box 516 St. Louis, Missouri 63166	
J. Gubser	1

	Copies
Lockheed Missiles and Space Co. Dept 81-12, Building 154 Box 504 Sunnyvale, California 94088 J. A. Bailie	1
Lockheed Missile and Space Co. Dept 52-30, Building 103 P.O. Box 504 Sunnyvale, California 94088	
Robert M. Beasley	1
Senior Staff Scientist	1
Lockheed Missile and Space Co. Technical Information Center 3251 Hanover Street Palo Alto, California 94034	1
Hughes Aircraft Company Missile System Division Canoga Park, California 91340 R. J. Oedy	1
Hughes Aircraft Company Culver City, California 90230 L. E. Gates, M.S. H-164	1
The Boeing Company P.O. Box 3999 Seattle, Washington 98124	
R. Blaisdell	1
K. Brettmann	1
Martin-Marietta Corporation P.O. Box 5837 Orlando, Florida 32805	
G. H. Sprague, Jr. (Director Aeromechanical Division)	1
G. Fotieo	1
Honeywell, Incorporated 600 2nd Street North Hopkins, Minnesota 55343 J. D. Brannan	1

	Copies
Fairchild Republic Co.	
Farmingdale, New York 11735	
Jim Clarke	1
Mail Stop Z-19A-08	1
Raytheon Co.	
Spencer Lab	
Burlington, Massachusetts 01803	
Doris A. Holden	1
Dept 7113	1
Total	150